A Tutorial on Evaluating Generative Models

Any sufficiently advanced technology is indistinguishable from magic. **Arthur C. Clarke**

Ali Borji Sep. 2021

Agenda

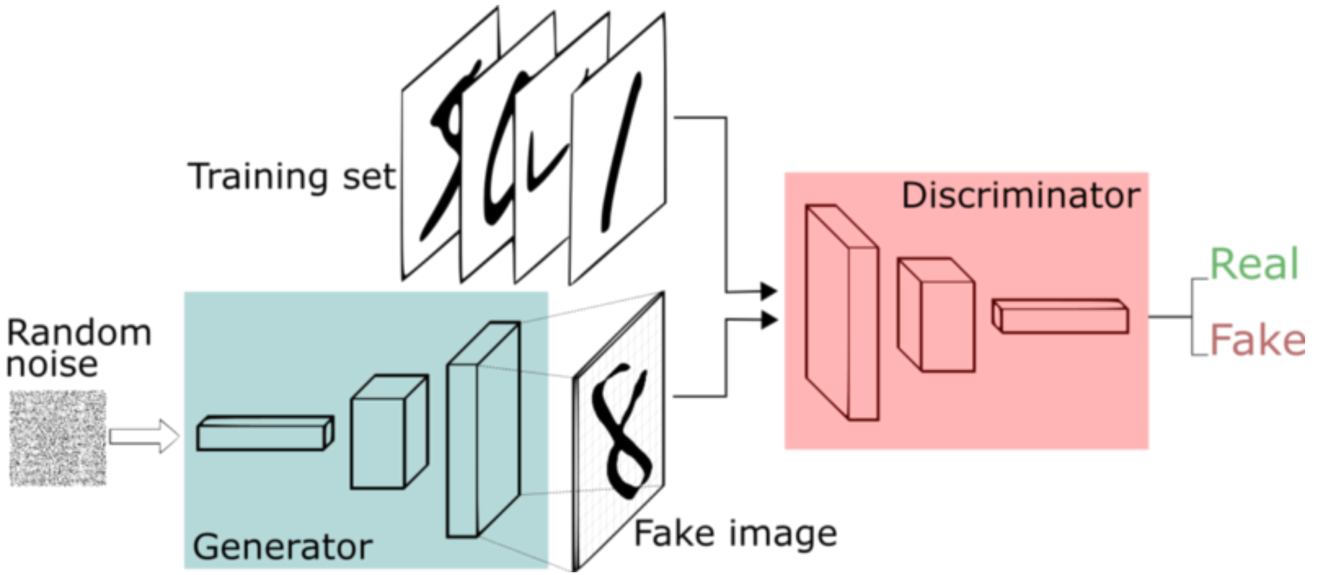
- Goal: An overview of major GAN evaluation measures and their pros and cons

- What will be covered:
  - What constitutes a good measure
  - **Quantitative** measures
  - **Qualitative** measures
  - Relation to deepfakes

- Emphasis on GANs
- Emphasis on evaluation, not modeling

Generative Adversarial Nets (GAN)

- Two networks play against each other:
  - **Generator**: generates images from noise (latent variables)
  - **Discriminator**: distinguishes between images generated by Generator and real images
- Eventually, the Generator learns to generate fake images that can be mistaken for real images
- GAN objective function:



$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

Goodfellow, et al, 2014

Generative Adversarial Nets (GAN)

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

  for k steps **do**

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
  - Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  end for

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

https://machinelearningmastery.com/how-to-code-the-generative-adversarial-network-training-algorithm-and-loss-functions/

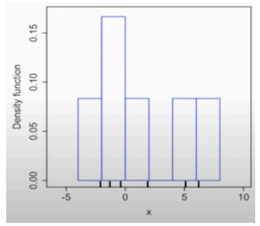
https://www.youtube.com/watch?v=Gib_kiXgnvA

Applications!

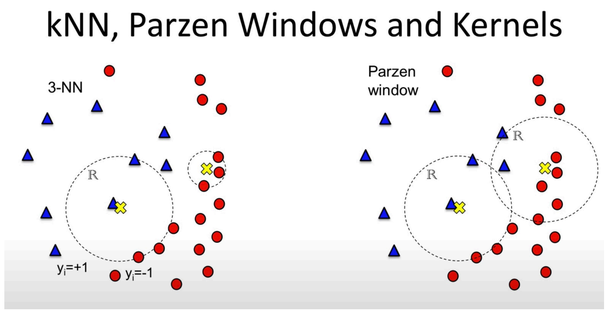
Density Estimation

Estimating an unknown probability density function given some data

- **Non-Parametric**

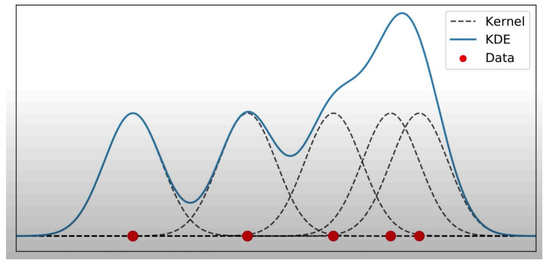


  - Histogram
  - k-Nearest Neighbor Density Estimation
  - Parzen Window Estimate



  - Kernel Density Estimation

$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N} K(x - x_i).$$



- **Parametric**

  - Maximum-Likelihood Estimation

$$p(\mathcal{D};\theta) = \prod_{i=1}^{n} p(x_i;\theta)$$

Sought: $p(x, y; \theta)$, where θ is fixed but unknown.

Performed: $\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\ p(\text{dataset}|\theta)$

In words: The ML solution seeks the solution, which is the best explanation of the dataset $X \in \mathcal{X}$ using the likelihood function, i.e. the class-conditional probability distribution

  - Bayesian Estimation

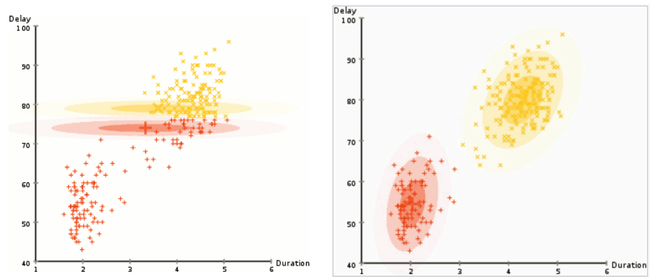
$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}$$

Sought: $p(x, y; \theta)$. θ is a random variable with the known prior $p(\theta)$.

In words: Bayesian method estimates the optimal parameter Θ of the given probability density, which maximizes the posterior probability distribution $p(\Theta|X)$.

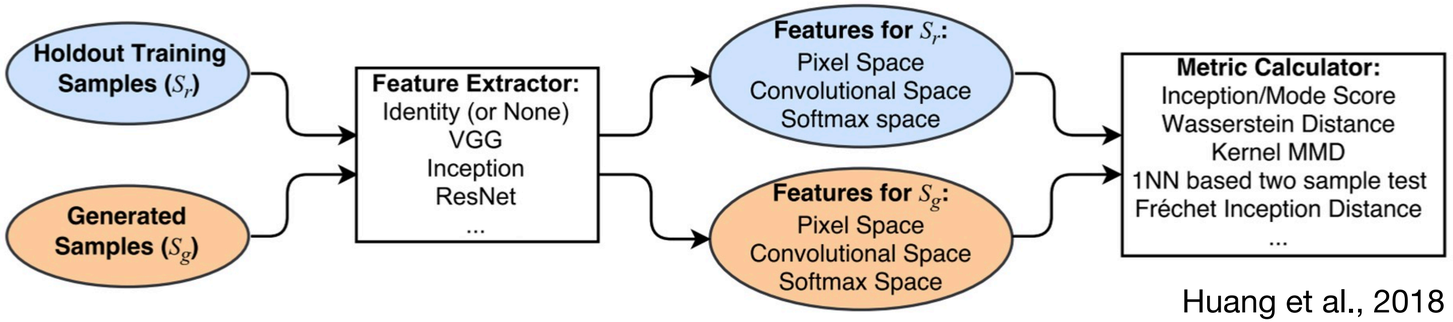
  - Expectation Maximization (EM)

An iterative method to find
ML & MAP



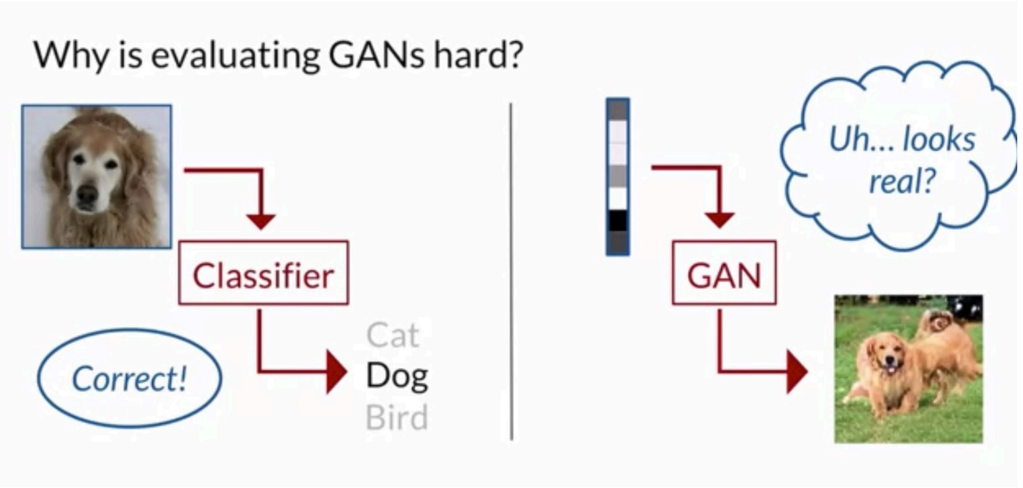
GAN Evaluation

- GAN is unsupervised learning
- GAN evaluation is hard, since there is no clear ground-truth!
- Is fooling a person enough?
- Many evaluation measures exist, some perform better than others (e.g., IS, FID, PR, PPL)
- Task dependency!



Huang et al., 2018

A schematic layout of the typical approach for sample based GAN evaluation. S_r and S_g represent real and generated samples, respectively.



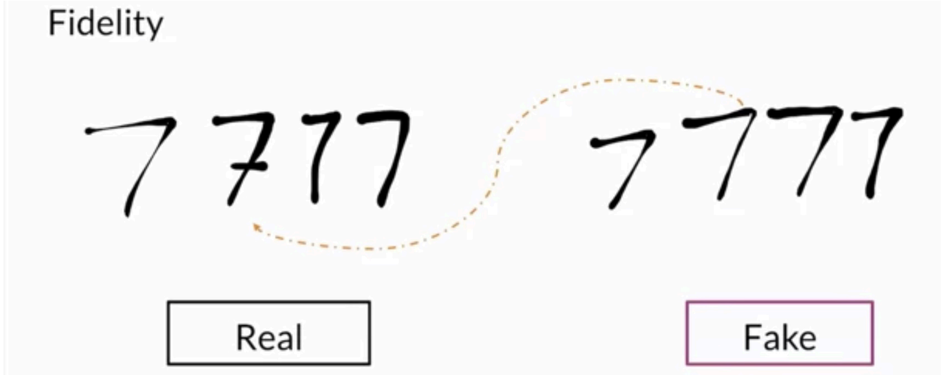
GAN Specialization, Coursera

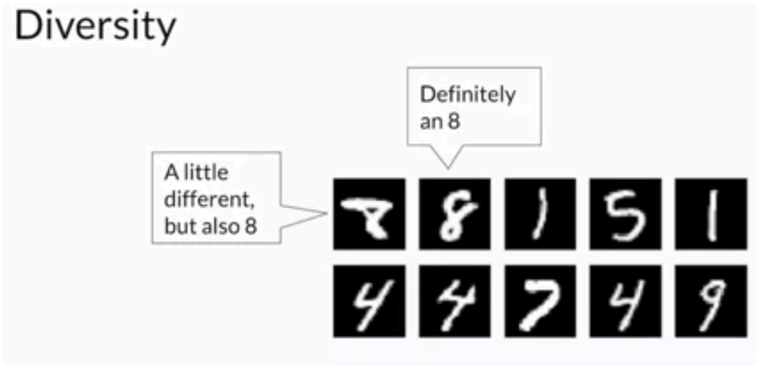
Fidelity vs. Diversity

Fidelity:
quality of images

Diversity:
variety of images

(Left) Available at: https://github.com/NVlabs/stylegan





Properties of a Good Measure

- Favors models that generate **high fidelity** samples
- Favors models that generate **diverse** samples
- Favors models with **disentangled latent spaces** as well as space continuity
- Has **well-defined bounds** (lower, upper, and chance)
- Is **sensitive to image distortions and transformations**
- Agrees with **human perceptual judgments and human rankings** of models
- Has **low sample and computational complexity**

Quantitative Measures

	Measure	Description
Quantitative	1. Average Log-likelihood [18, 22]	• Log likelihood of explaining realworld held out/test data using a density estimated from the generated data (*e.g.* using KDE or Parzen window estimation). $L = \frac{1}{N}\sum_i \log P_{model}(\mathbf{x}_i)$
	2. Coverage Metric [33]	• The probability mass of the true data "covered" by the model distribution $C := P_{data}(dP_{model} > t)$ with t such that $P_{model}(dP_{model} > t) = 0.95$
	3. Inception Score (IS) [3]	• KLD between conditional and marginal label distributions over generated data. $\exp\left(\mathbb{E}_{\mathbf{x}}\left[\mathbb{KL}\left(p\left(\mathrm{y} \mid \mathbf{x}\right) \parallel p\left(\mathrm{y}\right)\right)\right]\right)$
	4. Modified Inception Score (m-IS) [34]	• Encourages diversity within images sampled from a particular category. $\exp(\mathbb{E}_{\mathbf{x}_i}[\mathbb{E}_{\mathbf{x}_j}[(\mathbb{KL}(P(y\|\mathbf{x}_i)\|\|P(y\|\mathbf{x}_j))]])$
	5. Mode Score (MS) [35]	• Similar to IS but also takes into account the prior distribution of the labels over real data. $\exp\left(\mathbb{E}_{\mathbf{x}}\left[\mathbb{KL}\left(p\left(y \mid \mathbf{x}\right) \parallel p\left(y^{train}\right)\right)\right] - \mathbb{KL}\left(p\left(y\right) \parallel p\left(y^{train}\right)\right)\right)$
	6. AM Score [36]	• Takes into account the KLD between distributions of training labels vs. predicted labels, as well as the entropy of predictions. $\mathbb{KL}(p(y^{\text{train}}) \parallel p(y)) + \mathbb{E}_{\mathbf{x}}\left[H(y\|\mathbf{x})\right]$
	7. Fréchet Inception Distance (FID) [37]	• Wasserstein-2 distance between multi-variate Gaussians fitted to data embedded into a feature space $FID(r,g) = \|\mu_r - \mu_g\|_2^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{\frac{1}{2}})$
	8. Maximum Mean Discrepancy (MMD) [38]	• Measures the dissimilarity between two probability distributions P_r and P_g using samples drawn independently from each distribution. $M_k(P_r, P_g) = \mathbb{E}_{\mathbf{x},\mathbf{x}'\sim P_r}[k(\mathbf{x},\mathbf{x}')] - 2\mathbb{E}_{\mathbf{x}\sim P_r,\boldsymbol{y}\sim P_g}[k(\mathbf{x},\boldsymbol{y})] + \mathbb{E}_{\boldsymbol{y},\boldsymbol{y}'\sim P_g}[k(\boldsymbol{y},\boldsymbol{y}')]$
	9. The Wasserstein Critic [39]	• The critic (*e.g.* an NN) is trained to produce high values at real samples and low values at generated samples $\hat{W}(\mathbf{x}_{test}, \mathbf{x}_g) = \frac{1}{N}\sum_{i=1}^{N} \hat{f}(\mathbf{x}_{test}[i]) - \frac{1}{N}\sum_{i=1}^{N} \hat{f}(\mathbf{x}_g[i])$
	10. Birthday Paradox Test [27]	• Measures the support size of a discrete (continuous) distribution by counting the duplicates (near duplicates)
	11. Classifier Two Sample Test (C2ST) [40]	• Answers whether two samples are drawn from the same distribution (*e.g.* by training a binary classifier)
	12. Classification Performance [1, 15]	• An indirect technique for evaluating the quality of unsupervised representations (*e.g.* feature extraction; FCN score). See also the GAN Quality Index (GQI) [41].
	13. Boundary Distortion [42]	• Measures diversity of generated samples and covariate shift using classification methods.
	14. Number of Statistically-Different Bins (NDB) [43]	• Given two sets of samples from the same distribution, the number of samples that fall into a given bin should be the same up to sampling noise
	15. Image Retrieval Performance [44]	• Measures the distributions of distances to the nearest neighbors of some query images (*i.e.* diversity)
	16. Generative Adversarial Metric (GAM) [31]	• Compares two GANs by having them engaged in a battle against each other by swapping discriminators or generators. $p(\mathbf{x}\|y=1; M_1^{'})/p(\mathbf{x}\|y=1; M_2^{'}) = \big(p(y=1\|\mathbf{x}; D_1)p(\mathbf{x}; G_2)\big)/\big(p(y=1\|\mathbf{x}; D_2)p(\mathbf{x}; G_1)\big)$
	17. Tournament Win Rate and Skill Rating [45]	• Implements a tournament in which a player is either a discriminator that attempts to distinguish between real and fake data or a generator that attempts to fool the discriminators into accepting fake data as real.
	18. Normalized Relative Discriminative Score (NRDS) [32]	• Compares n GANs based on the idea that if the generated samples are closer to real ones, more epochs would be needed to distinguish them from real samples.
	19. Adversarial Accuracy and Divergence [46]	• Adversarial Accuracy. Computes the classification accuracies achieved by the two classifiers, one trained on real data and another on generated data, on a labeled validation set to approximate $P_g(y\|\mathbf{x})$ and $P_r(y\|\mathbf{x})$. Adversarial Divergence: Computes $\mathbb{KL}(P_g(y\|\mathbf{x}), P_r(y\|\mathbf{x}))$
	20. Geometry Score [47]	• Compares geometrical properties of the underlying data manifold between real and generated data.
	21. Reconstruction Error [48]	• Measures the reconstruction error (*e.g.* L_2 norm) between a test image and its closest generated image by optimizing for z (*i.e.* $min_{\mathbf{z}}\|\|G(\mathbf{z}) - \mathbf{x}^{(test)}\|\|^2$)
	22. Image Quality Measures [49, 50, 51]	• Evaluates the quality of generated images using measures such as SSIM, PSNR, and sharpness difference
	23. Low-level Image Statistics [52, 53]	• Evaluates how similar low-level statistics of generated images are to those of natural scenes in terms of mean power spectrum, distribution of random filter responses, contrast distribution, etc.
	24. Precision, Recall and F_1 score [23]	• These measures are used to quantify the degree of overfitting in GANs, often over toy datasets.

+

Perceptual Path Length (PPL)
Classification Accuracy Scores (CAS)
Measures that Probe Generalization
Spectral Methods

Pros and Cons of GAN Evaluation Measures, Borji, 2019
Pros and Cons of GAN Evaluation Measures: New Developments, Borji, 2021

Average Log-likelihood

- Equivalent to Kullback-Leibler divergence

$$D_{KL}[P(x|\theta^*) \,\|\, P(x|\theta)] = \mathbb{E}_{x\sim P(x|\theta^*)}\left[\log \frac{P(x|\theta^*)}{P(x|\theta)}\right]$$

$$= \mathbb{E}_{x\sim P(x|\theta^*)}\left[\log P(x|\theta^*) - \log P(x|\theta)\right]$$

$$= \mathbb{E}_{x\sim P(x|\theta^*)}\left[\log P(x|\theta^*)\right] - \mathbb{E}_{x\sim P(x|\theta^*)}\left[\log P(x|\theta)\right]$$

- Estimating likelihood in higher dimensions is not feasible
- Generally uninformative about the quality of samples and vice versa
- A mixture of Gaussians with training images as the means will generate great samples but will still have very poor log-likelihood
- Produces rankings different from other measures

.

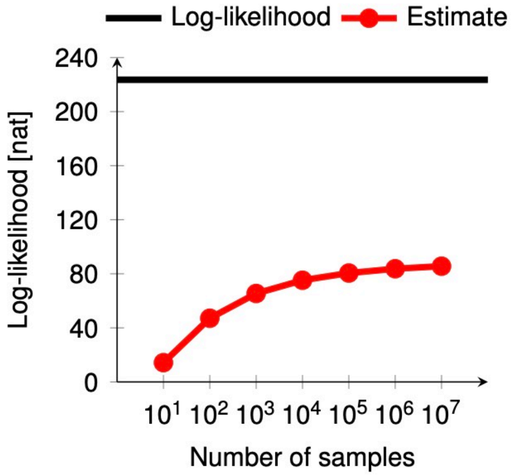


Figure 3: Parzen window estimates for a Gaussian evaluated on 6 by 6 pixel image patches from the CIFAR-10 dataset. Even for small patches and a very large number of samples, the Parzen window estimate is far from the true log-likelihood.

Model	Parzen est. [nat]
Stacked CAE	121
DBN	138
GMMN	147
Deep GSN	214
Diffusion	220
GAN	225
True distribution	**243**
GMMN + AE	282
k-means	313

Table 1: Using Parzen window estimates to evaluate various models trained on MNIST, samples from the true distribution perform worse than samples from a simple model trained with k-means.

Theis et al., 2016

Inception Score

$$\mathrm{IS} = \exp(\mathbb{E}_{x \sim p_g} D_{KL}(p(y \mid x) \| p(y)))$$

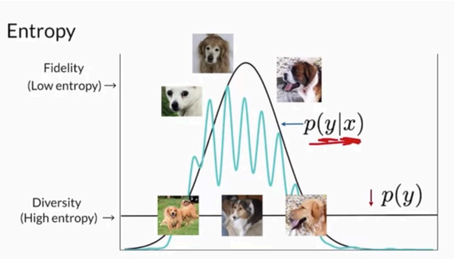
KL Divergence

- Based on the Inception-v3 predictions
- Has well-defined bounds, low is 1 and max is number of classes. IS over real images can serve as the upper bound
- Shows a reasonable correlation with the quality and diversity of generated images.
- Can be easily gamed (by memory GAN)
- Only considers fake images
- Can not detect mode collapse!
- Biased towards ImageNet/objects & Inception model
- When IS is bad it is usually because:
    - Both terms have low entropy i.e. are picky (no diversity)
    - Both terms have high entropy (no fidelity)

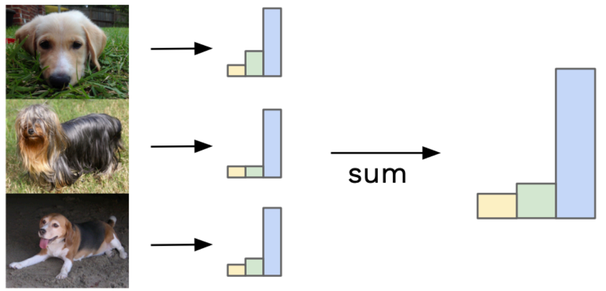
$$D_{KL}(p(y|x)\|p(y)) = p(y|x)\log\left(\frac{p(y|x)}{p(y)}\right)$$

Conditional distribution (fidelity)

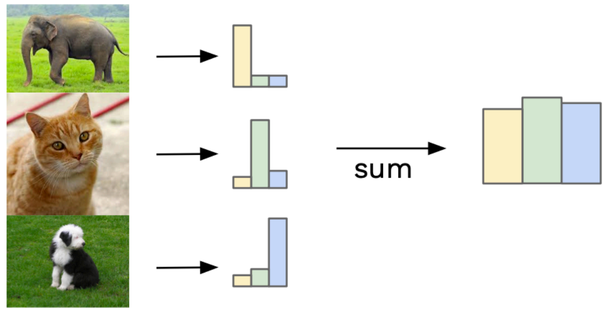
Marginal distribution (diversity)

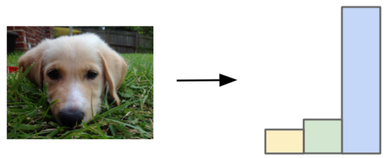


Similar labels sum to give focussed distribution

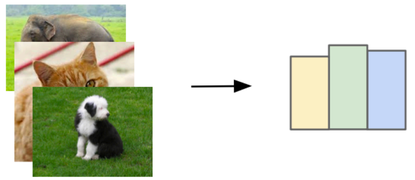


Different labels sum to give uniform distribution





Ideal label distribution

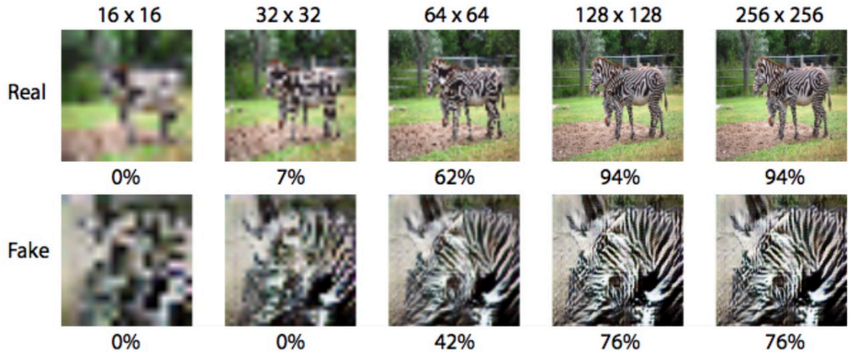


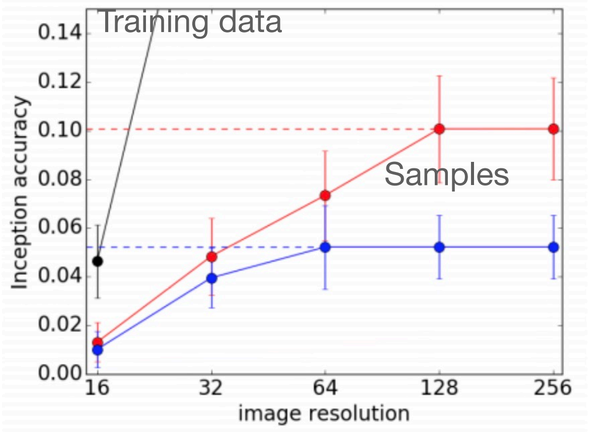
Ideal marginal distribution

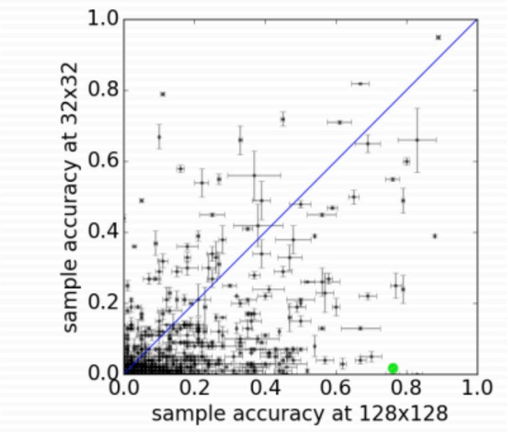
Salimans et al., 2017
GAN Specialization, Coursera
https://medium.com/octavian-ai/a-simple-explanation-of-the-inception-score-372dff6a8c7a

IS (cnt'd)

Sensitivity of IS to image resolution
(IS is shown below each image)







- **IS Variants**
  - Original

$$\exp\left(\mathbb{E}_{\mathbf{x}}\left[\mathbb{KL}\left(p\left(\mathrm{y}\mid\mathbf{x}\right)\parallel p\left(\mathrm{y}\right)\right)\right]\right)=\exp\left(H(y)-\mathbb{E}_{\mathbf{x}}\left[H(y|\mathbf{x})\right]\right),$$

  - Modified Inception Score (m-IS)

$$\exp(\mathbb{E}_{\mathbf{x}_i}[\mathbb{E}_{\mathbf{x}_j}[(\mathbb{KL}(P(y|\mathbf{x}_i)||P(y|\mathbf{x}_j))]]),$$

  - Mode score

$$\exp\left(\mathbb{E}_{\mathbf{x}}\left[\mathbb{KL}\left(p\left(y\mid\mathbf{x}\right)\parallel p\left(y^{train}\right)\right)\right]-\mathbb{KL}\left(p\left(y\right)\parallel p\left(y^{train}\right)\right)\right),$$

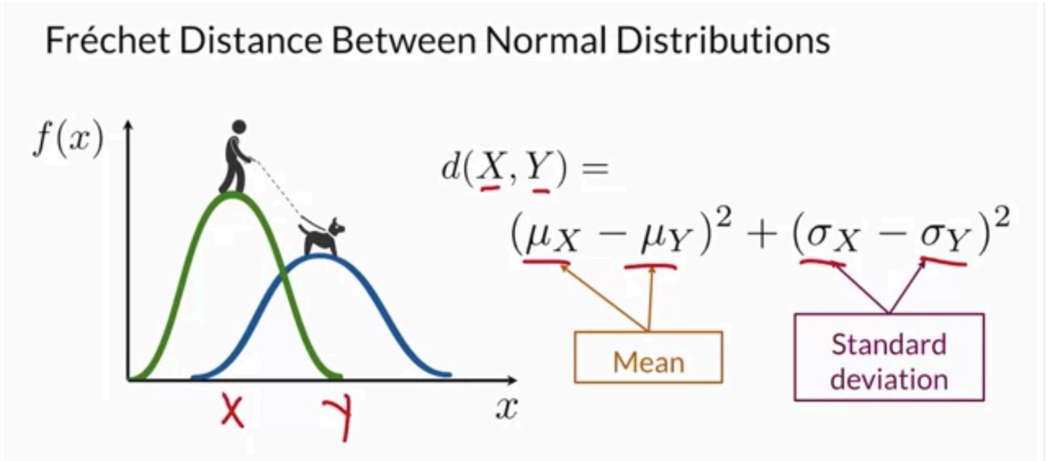
  - AM score

$$\mathbb{KL}(p(y^{\mathrm{train}})\parallel p(y))+\mathbb{E}_{\mathbf{x}}\left[H(y|\mathbf{x})\right].$$

  - Unbiased IS
  - …

Fréchet Inception Distance (FID)

- Based on the Inception embeddings (2048 D)
- Assumes that features follow a multivariate Gaussian distribution to make computation easy (a.k.a Wasserstein-2 distance)
- Lower is better. Lower bound = 0
- Can be used to monitor the progress during training
- Is sensitive to mode collapse
- Requires a large sample size to be reliable (usually ~50K fakes, ~50K real images) —> Slow to run
- Dependency on the amount of data; typically FID is lower for larger sample sizes, thus may fool u that the model is better
- Only considers two moments of distributions
- Biased towards ImageNet/objects & Inception model



Univariate Normal Fréchet Distance =

$$(\mu_X - \mu_Y)^2 + (\sigma_X - \sigma_Y)^2$$

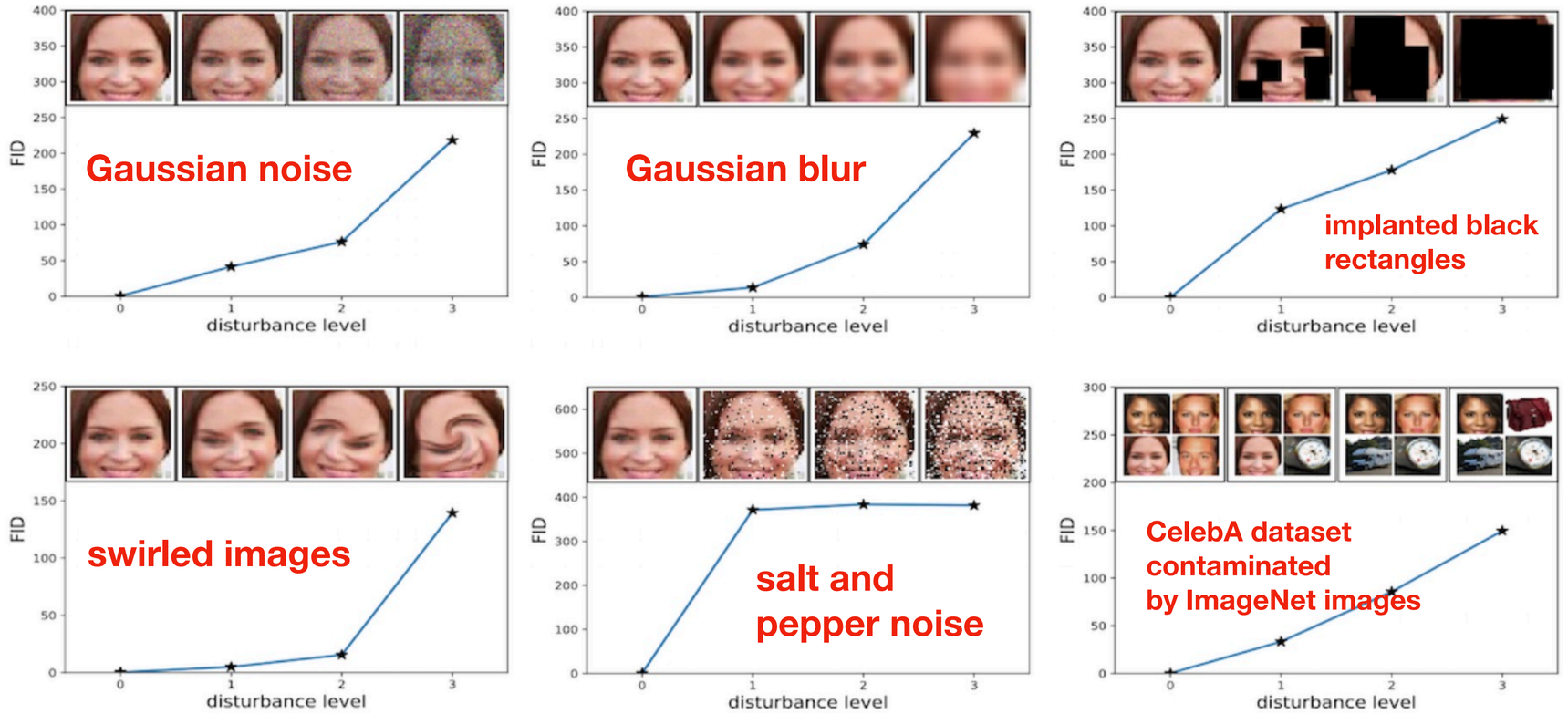
Multivariate Normal Fréchet Distance =

$$\|\mu_X - \mu_Y\|^2 + \mathrm{Tr}\left(\Sigma_X + \Sigma_Y - 2\sqrt{\Sigma_X \Sigma_Y}\right)$$

Heusel et al., 2017
GAN Specialization, Coursera

FID (cnt'd)

Sensitivity of FID to image resolution



- **FID Variants**
  - Unbiased FID
  - Memorization-informed FID (MiFID
  - Fast FID
  - Class-aware FID (CAFD)
  - Conditional FID
  - Spatial FID (sFID)
  - …

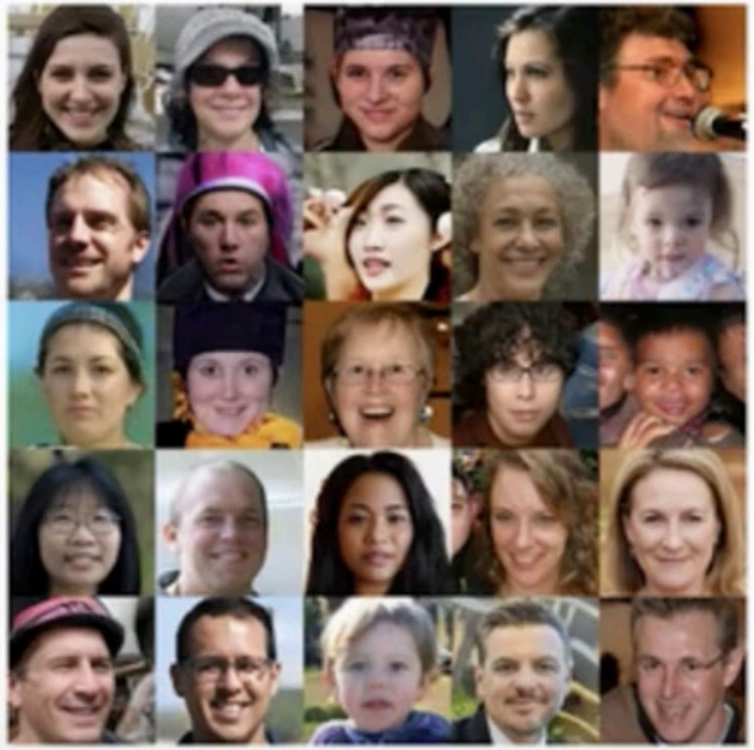
Heusel et al., 2017

https://www.youtube.com/watch?v=_XwsGkryVpk&t=837s

Does FID encode fidelity?

FID=40.8
Less realistic

<



FID=8.3
More realistic

- Yes, it does.

Does FID encode diversity?

FID=16.1
Less diverse

<

FID=8.3
More diverse

- Yes, it does.

Does FID encode fidelity & diversity?

FID=7.8
Less diverse
More realistic

?

FID=8.3
More diverse
Less realistic

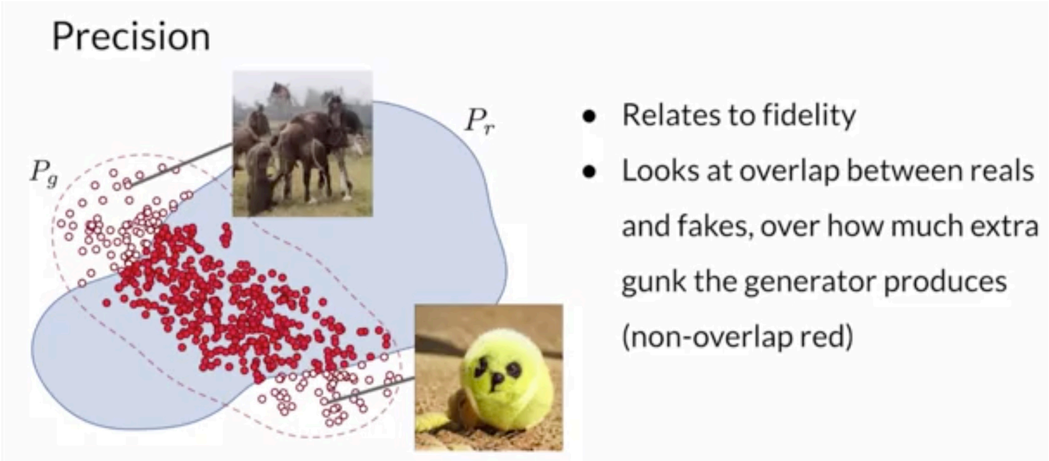
- FID confounds fidelity and diversity.
- We need separate metrics for them for precise diagnosis.

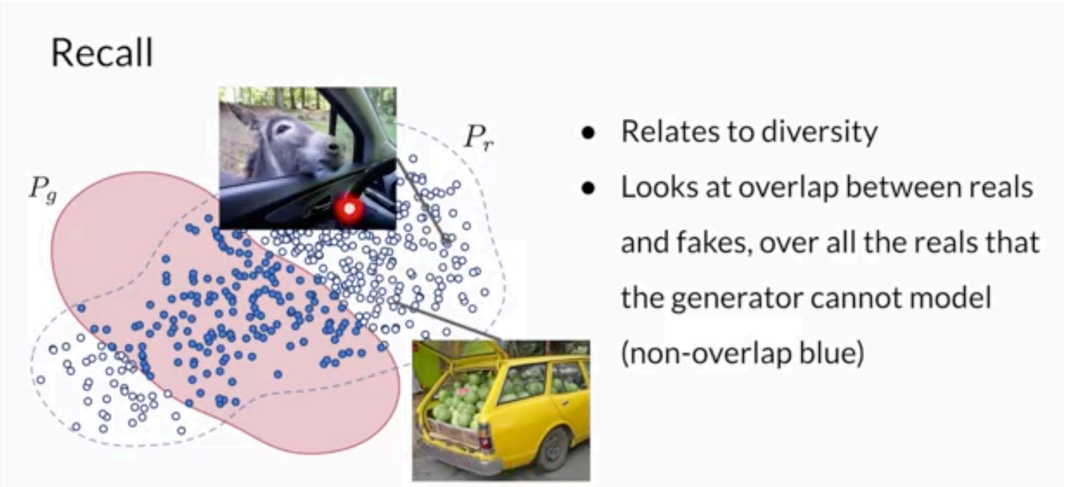
Precision and Recall (PR)

- **Precision** tp / (tp + fp)
  - Among fake samples, which proportion is close to real?
  ("Close to" is defined in terms of the kNN radii)

- **Recall** tp / (tp + fn)
  - Among real samples, which proportion is close to fake?

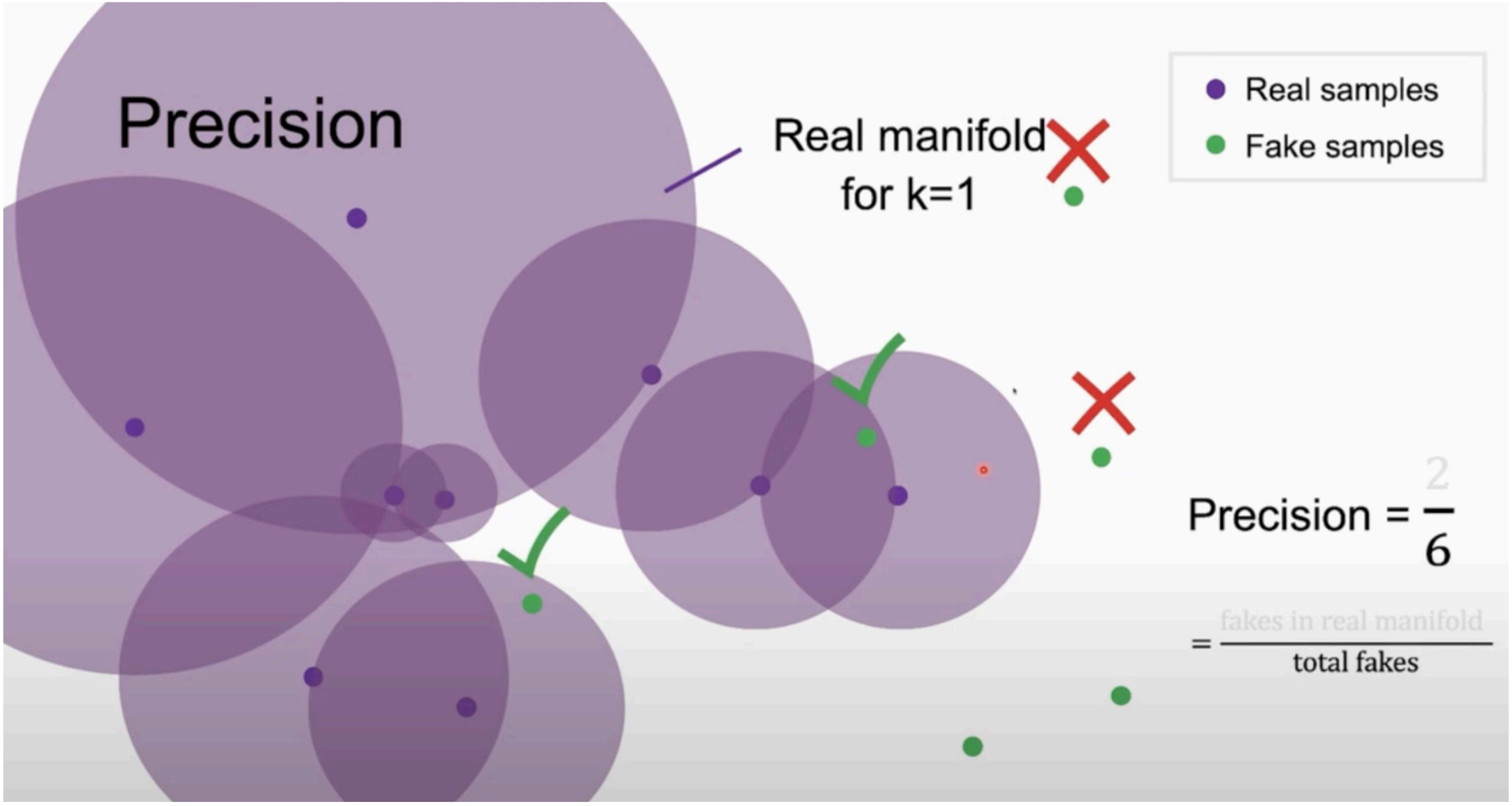
- **Variants:** Density & Coverage, Alpha Precision & Recall, …



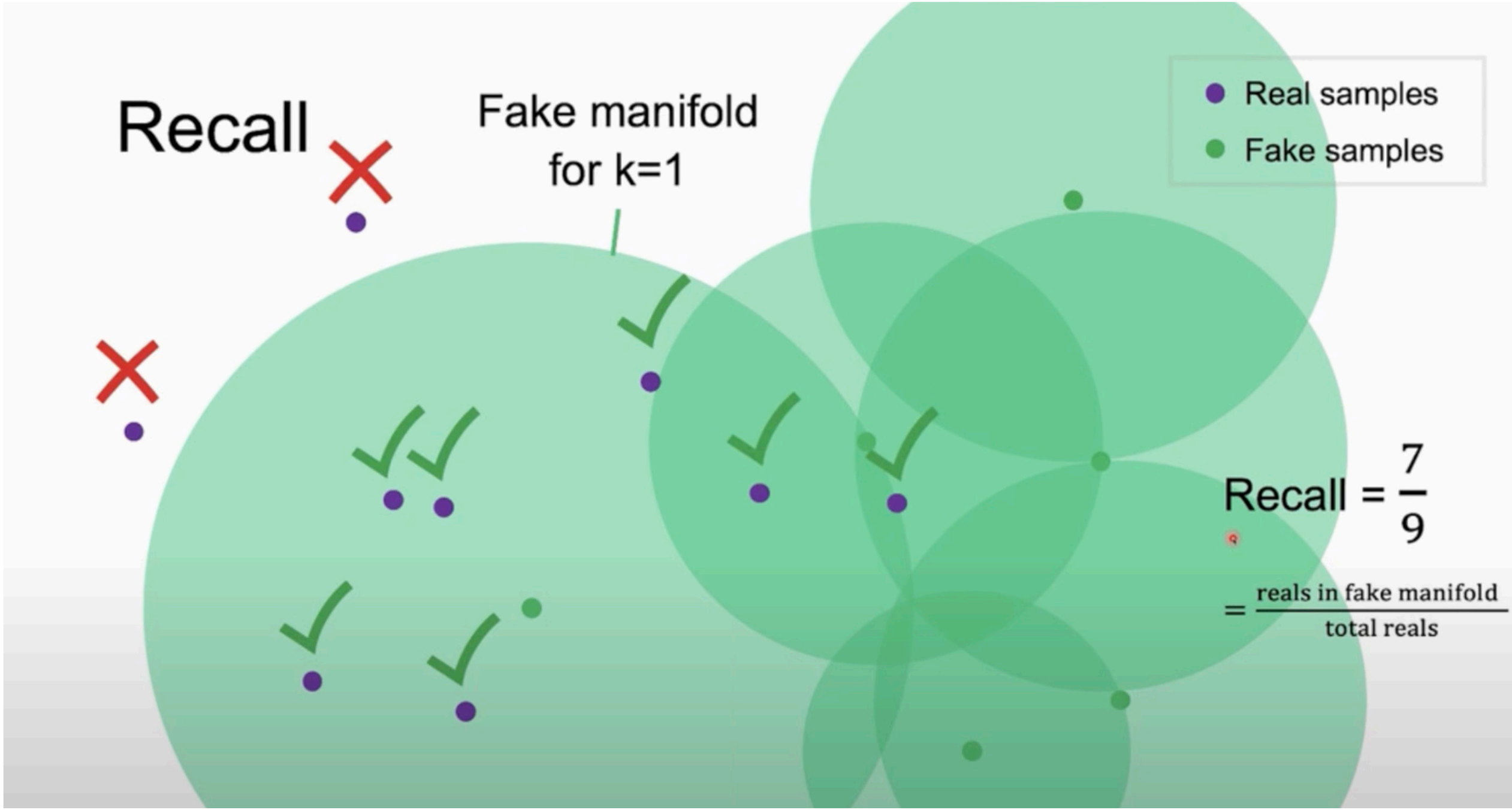


Sajjadi et al., 2018

GAN Specialization, Coursera



https://www.youtube.com/watch?v=_XwsGkryVpk&t=805s


Recall
Fake manifold
for k=1
Real samples
Fake samples
Recall = 7/9
= reals in fake manifold / total reals

Perceptual Path Length (PPL)

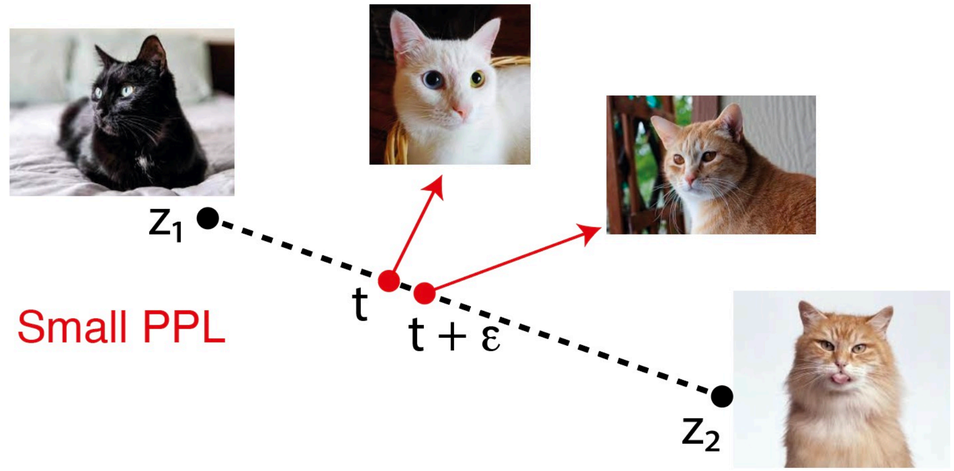
- Measures whether and how much the latent space of a generator is entangled
- A less curved latent space should result in perceptually smoother transition than a highly curved latent space
- Small changes in the latent vector Z should not result in too dramatic of the changes in the generated image
- PPL is the empirical mean of the perceptual difference between consecutive images in the latent space Z, over all possible endpoints:

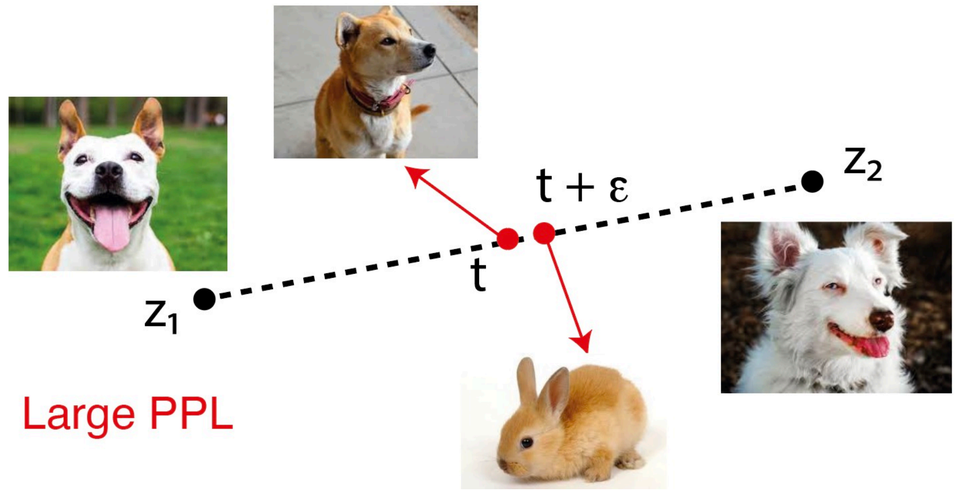
$$l_{\mathcal{Z}} = \mathbb{E}[\frac{1}{\epsilon^2} d\big(G(\text{slerp}(\mathbf{z}_1, \mathbf{z}_2;\, t)), G(\text{slerp}(\mathbf{z}_1, \mathbf{z}_2;\, t+\epsilon))\big)\Big],$$

where z1, z2 ~ P(z), t ~ U(0, 1), G is the generator, and d(·, ·) is the perceptual distance between the resulting images. slerp* denotes the spherical linear interpolation

---

See https://tiborstanko.sk/lerp-vs-slerp.html





Karras et al., 2019

PPL (cnt’d)

(a) Low PPL scores (b) High PPL scores

Correlation between perceptual path length and image quality (generated by StyleGAN). The lower the PPL, the better.

The LPIPS (Learned Perceptual Image Patch Similarity) measures the perceptual distance between two images. It is a weighted L2 difference between two embeddings, where the weights are learned to make the metric agree with human similarity judgments

Zhang et al., 2018

Karras et al., 2019

Model 1: FID = 8.53, P = 0.64, R = 0.28, PPL = 924

Model 2: FID = 8.53, P = 0.62, R = 0.29, PPL = 387

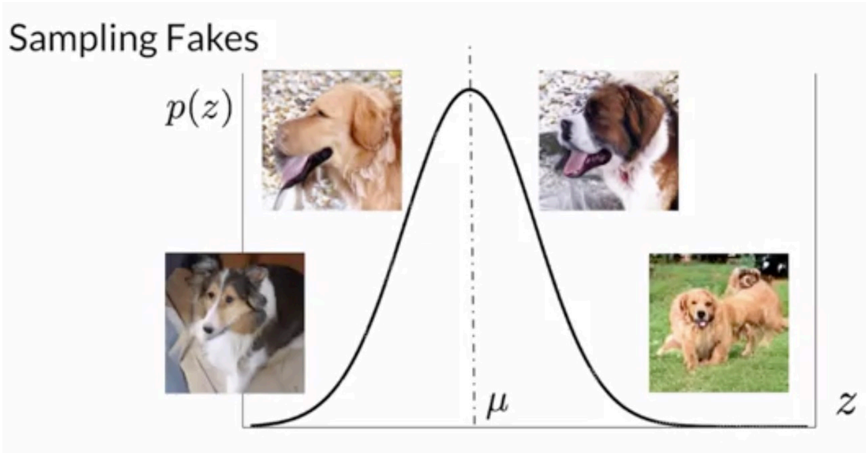
Synthesized samples from two generative models trained on LSUN CAT without truncation.

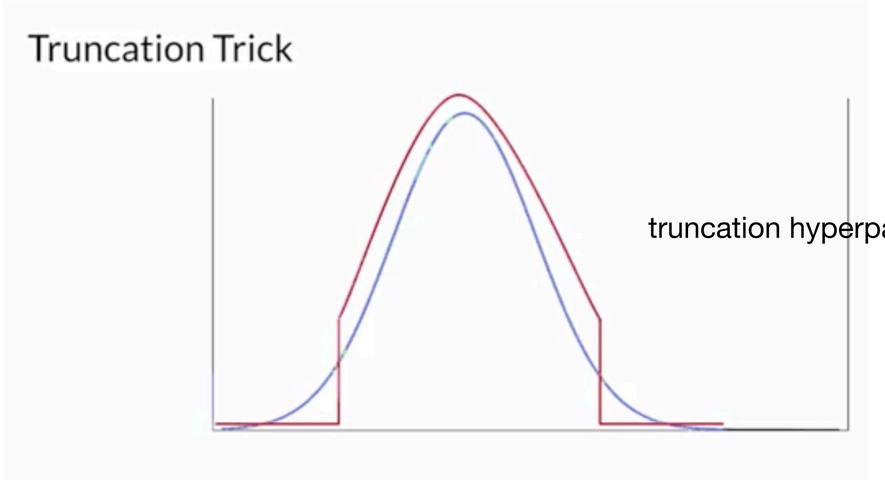
FID, precision (P), and recall (R) are similar for the two models, even though the latter produces cat-shaped objects more often. Perceptual path length (PPL) shows a clear preference for model 2.

Karras et al., 2019

Sampling matters!

- **Truncation**: Drawing latent vectors from a truncated or shrunk sampling space to improve average image quality, at the expense of loosing diversity
- Sample from a truncated normal where values which fall outside a range are resampled to fall inside that range
- Is usually done after the model has been trained and it broadly trades off fidelity and diversity
- Truncate more for higher fidelity, lower diversity
- Truncation helps precision (in PR), hurts recall and also FID!
- Trade-off can be tuned for downstream applications





Classifier Two-sample Tests (C2ST)

- Tests whether two samples are drawn from the same distribution
- Assume we have access to two sets of samples:

$$S_P = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\} \sim P^n(X) \text{ and } S_Q = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n\} \sim Q^n(Y)$$

- Test whether the null hypothesis

$H_0 : P = Q$ is true following steps (a) to (e):

- In principle, any binary classifier can be adopted for computing C2ST

(a) Construct the following dataset

$$\mathcal{D} = \{(\mathbf{x}_i, 0)\}_{i=1}^n \cup \{(\boldsymbol{y}_i, 1)\}_{i=1}^n =: \{(\mathbf{z}_i, l_i)\}_{i=1}^{2n}.$$

(b) Randomly shuffle $\mathcal{D}$, and split it into two disjoint *training* and *testing* subsets $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$, where $\mathcal{D} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{test}}$ and $n_{\text{test}} := |\mathcal{D}_{\text{test}}|$.

(c) Train a binary classifier $f : \mathcal{X} \to [0, 1]$ on $\mathcal{D}_{\text{train}}$. In the following, assume that $f(\mathbf{z}_i)$ is an estimate of the conditional probability distribution $p(l_i = 1|\mathbf{z}_i)$.

(d) Calculate the classification accuracy on $\mathcal{D}_{\text{test}}$:

$$\hat{t} = \frac{1}{n_{\text{test}}} \sum_{(\mathbf{z}_i, l_i) \in \mathcal{D}_{\text{test}}} \mathbb{I}\left[\mathbb{I}\left(f(\mathbf{z}_i) > \frac{1}{2}\right) = l_i\right] \tag{10}$$

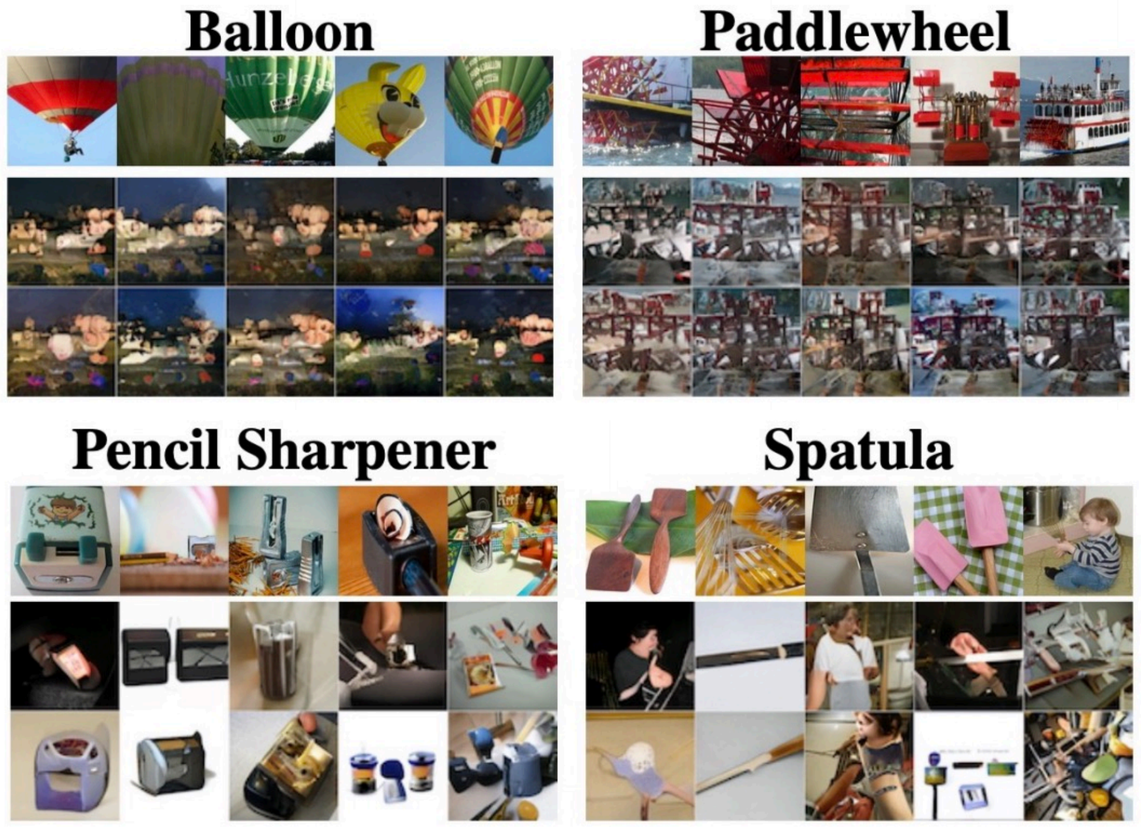
as the *C2ST statistic*, where $\mathbb{I}$ is the indicator function. The intuition here is that if $P = Q$, the test accuracy in Eq. 10 should remain near chance-level. In contrast, if binary classifier performs better than chance then it implies that $P \neq Q$.

(e) To accept or reject the null hypothesis, compute a p-value using the null distribution of the C2ST.

Lopez-Paz & Oquab, 2016

Classification Accuracy Score (CAS)

- If a generative model is learning the data distribution in a perceptually meaningful space then it should perform well in downstream tasks.
- **Train an image classifier using only synthetic data and use it to predict labels of real images in the test set**
- CAS automatically identifies particular classes for which generative models fail to capture the data distribution
- **Ravuri & Vinyals, 2019:**
  - IS and FID are neither predictive of CAS, nor useful when evaluating non-GAN models
  - Using a state-of-the-art GAN accuracy decreases
- Similarly, FCN score by Isola et al. 2017



CAS can identify classes for which a generative model, here BigGAN-deep, fails to capture the data distribution (top row: real images, bottom two rows: generated samples).

Ravuri & Vinyals, 2019

Birthday Paradox Test

- Approximates the **support*** size of a discrete distribution
- It works as follows:
  - Pick a sample of size **S** from the generated distribution
  - Use an automated measure of image similarity to flag the k (e.g. k = 20) most similar pairs in the sample
  - Visually inspect the flagged pairs and check for (near) duplicates
  - Repeat
- If this test reveals that samples of size s have duplicate images with good probability, then suspect that the distribution has support size about **s^2**

With k as small as 23, with probability > 50%, at least two people in the room have the same birthday.

Example:

With probability > 50%, a batch of about 400 samples generated from CelebA dataset contains at least one pair of duplicates for DCGAN, thus leading to support size of 400^2

* The support of a real-valued function f is the subset of the domain containing those elements which are not mapped to zero.

Arora & Zhang, 2017

https://www.youtube.com/watch?v=QrwV6fJKBi8

Generative Adversarial Metric (GAM)

- Engaging GANs in a battle against each other by swapping discriminators or generators across the two models
- Measures the relative performance of two GANs by measuring the likelihood ratio of the two models

$$\frac{p(\mathbf{x}|y=1;M_1')}{p(\mathbf{x}|y=1;M_2')} = \frac{p(y=1|\mathbf{x};D_1)p(\mathbf{x};G_2)}{p(y=1|\mathbf{x};D_2)p(\mathbf{x};G_1)},$$

- **M1 is better than M2 if G1 fools D2 more than G2 fools D1 (and vice versa)**
- GAM suffers from two main caveats:
  - It has a constraint where the two discriminators must have an approximately similar performance on a calibration dataset
  - It is expensive to compute because it has to be computed for all pairs of models

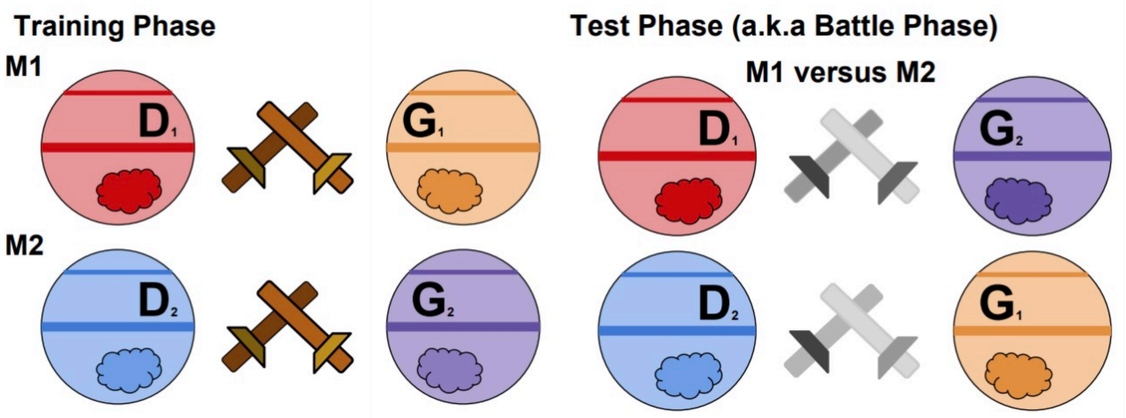


Illustration of the Generative Adversarial Metric (GAM). During the training phase, G1 and G2 compete with D1 and D2, respectively. At test time, model M1 plays against M2 by having G1 try to fool D2, and vice-versa. M1 is better than M2 if G1 fools D2 more than G2 fools D1 (and vice versa)

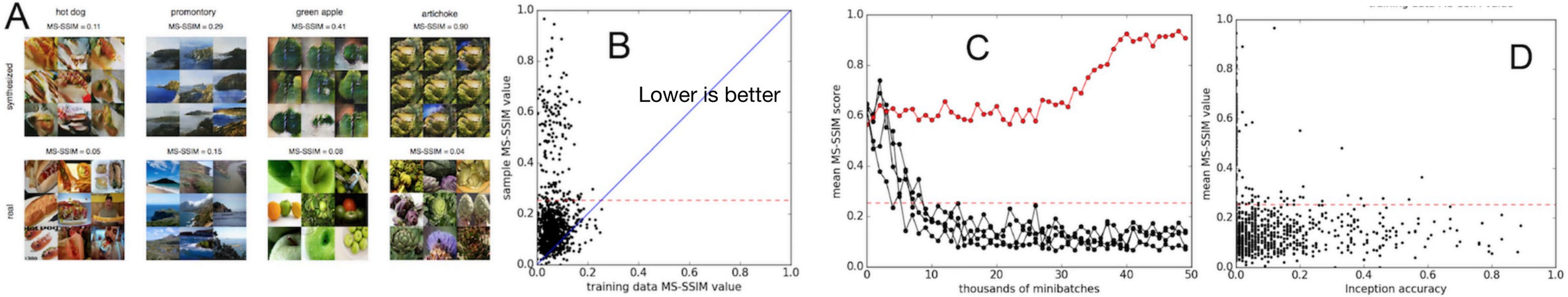
Im et al., 2016

Image Quality Measures

SSIM, PSNR and Sharpness Difference

See also LPIPS by Zhang et al., 2018

- Geared more towards fidelity

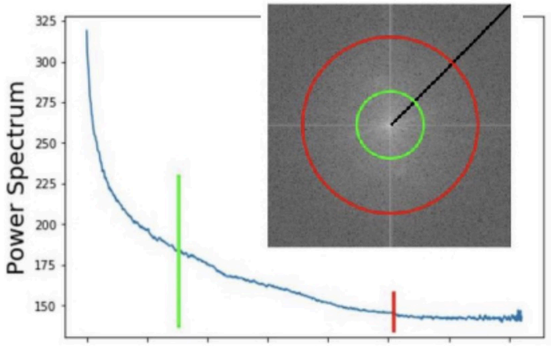


A) MS-SSIM scores for generated samples (top row) and training samples (bottom row).
B) The mean MS-SSIM scores between pairs of images within a given class of the ImageNet dataset versus AC-GAN samples. The horizontal red line marks the maximum MS-SSIM across all ImageNet classes (over training data). Each data point represents the mean MS-SSIM value for samples from one class.
C) Intra-class MS-SSIM for five ImageNet classes throughout a training run. Here, decreasing trend means successful training (black lines) whereas increasing trend indicates that the generator ‘collapses’ (red line).
D) Comparison of Inception score vs. MS-SSIM for all 1000 ImageNet classes

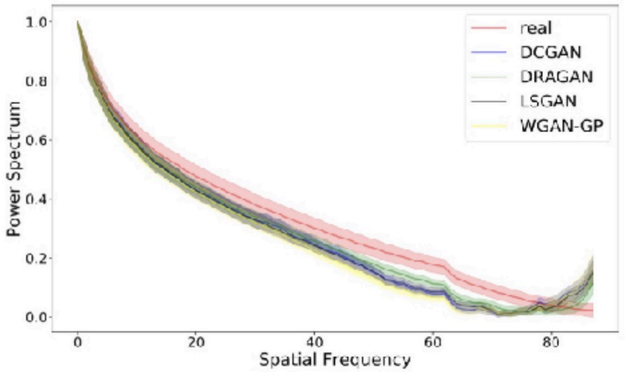
Odena et al., 2016

Spectral Methods

- Test whether generators are able to correctly approximate the spectral distributions of real data
- Up-scaling operations commonly used in GANs (e.g. up-convolutions) alter the spectral properties of the images causing high frequency distortions in the output
- Can be used as a regularization term to improve GANs
- Can be used to build detectors to identify fake images and videos



An example of azimuthal integral for an image



Statistics (mean and variance) after azimuthal integration over the power-spectrum of real and GAN generated images over the CelebA dataset

The left-most panel shows that mean DCT spectrum of the FFHQ data set with a sample from this dataset next to it. The right-most side shows he mean DCT spectrum of a data set sampled from StyleGAN trained on FFHQ, with a generated face to its left. Results are averaged over 10,000 images.

Durall et al., 2020
Frank et al., 2020

Qualitative Measures

Qualitative	1. Nearest Neighbors	• To detect overfitting, generated samples are shown next to their nearest neighbors in the training set
	2. Rapid Scene Categorization [18]	• In these experiments, participants are asked to distinguish generated samples from real images in a short presentation time (*e.g.* 100 ms); *i.e.* real v.s fake
	3. Preference Judgment [54, 55, 56, 57]	• Participants are asked to rank models in terms of the fidelity of their generated images (*e.g.* pairs, triples)
	4. Mode Drop and Collapse [58, 59]	• Over datasets with known modes (*e.g.* a GMM or a labeled dataset), modes are computed as by measuring the distances of generated data to mode centers
	5. Network Internals [1, 60, 61, 62, 63, 64]	• Regards exploring and illustrating the internal representation and dynamics of models (*e.g.* space continuity) as well as visualizing learned features

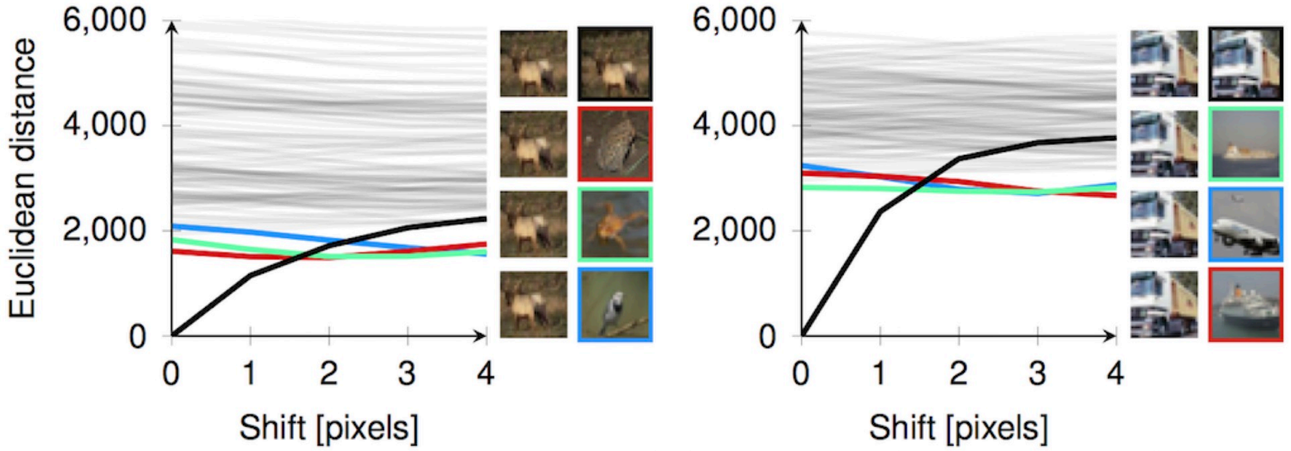
+

Human Eye Perceptual Evaluation (HYPE)
GAN Dissection
GAN Steerability

Pros and Cons of GAN Evaluation Measures, Borji, 2019
Pros and Cons of GAN Evaluation Measures: New Developments, Borji, 2021

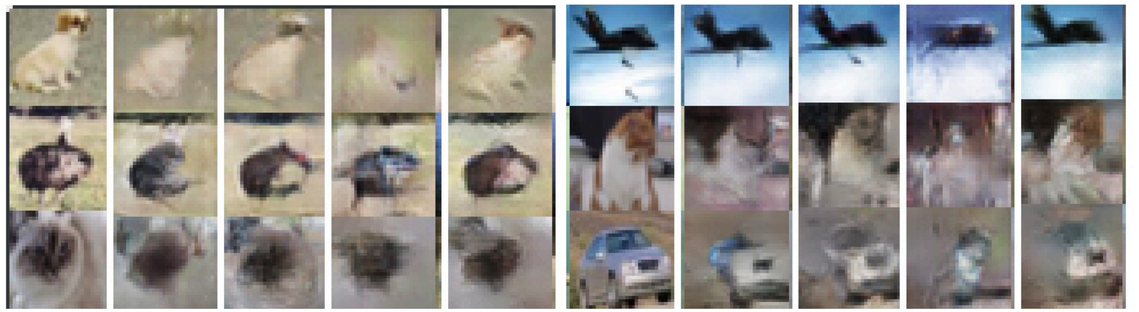
Nearest Neighbors

- Helps detect overfitting
- Nearest neighbors are typically determined based on the Euclidean distance which is very sensitive to minor perceptual perturbations
- It is trivial to generate samples that are visually almost identical to a training image, but have large Euclidean distances with it
- A model that stores (transformed) training images (i.e. memory GAN) can trivially pass the nearest-neighbor overfitting test.
- These problems can be alleviated by choosing nearest neighbors based on perceptual measures, and by showing more than one nearest neighbor



Small changes to an image can lead to large changes in Euclidean distance. The left column shows a query image shifted 1 to 4 pixels (top to bottom). The right column shows the corresponding nearest neighbor from the training set. The gray lines indicate Euclidean distance of the query image to 100 randomly picked images from the training set.

Theis et al., 2016



Generated samples nearest to real images from CIFAR-10. In each of the two panels, the first column shows real images, followed by the nearest image generated by DCGAN, ALI, Unrolled GAN, and VEEGAN, respectively.

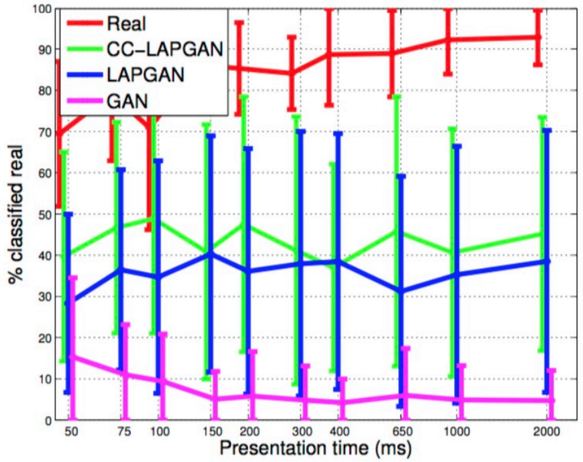
Srivastava et al., 2017

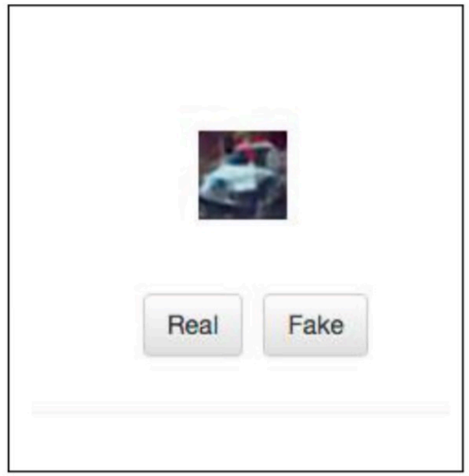
Rapid Scene Categorization

- A“Turing-like” test
- Inspired by prior studies who have shown that humans are capable of reporting certain characteristics of scenes in a short glance (e.g. scene category, visual layout)
- Experimental conditions are hard to control in crowd-sourced platforms (e.g. presentation time, screen size, subject’s distance to the screen, subjects’ motivations, etc)
- Expensive!
- Biased toward fidelity rather than diversity
- Biased towards models that overfit to training data

.

See if you can spot a problem with this setup!?





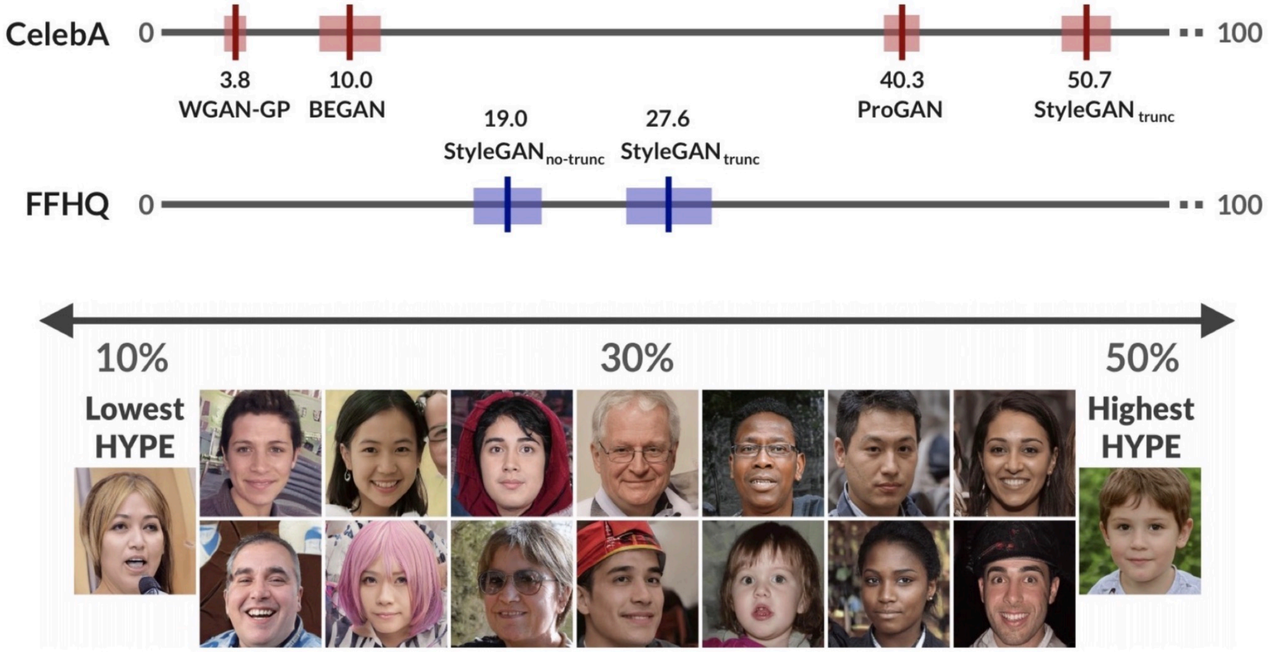
Left: Human evaluation of real CIFAR10 images (red) and samples from Goodfellow et al. (magenta), and LAPGAN and a class conditional LAPGAN (green)

Right: The user-interface presented to the subjects.

Denton et al., 2015

Human Eye Perceptual Evaluation (HYPE)

- Tests how realistic generated images look to the human eye
- A human-in-the-loop measure, crowd sourcing
- Is grounded in psychophysics research on visual perception
- **Two variants:**
  - HYPE time: Adaptive time constraints to determine the threshold at which a model's outputs appear real (e.g. 250 ms)
  - HYPE infinity: Measures human error rate on fake and real images without time constraints
- Is expensive and is hard to scale, although it might be possible to train models to approximate human response



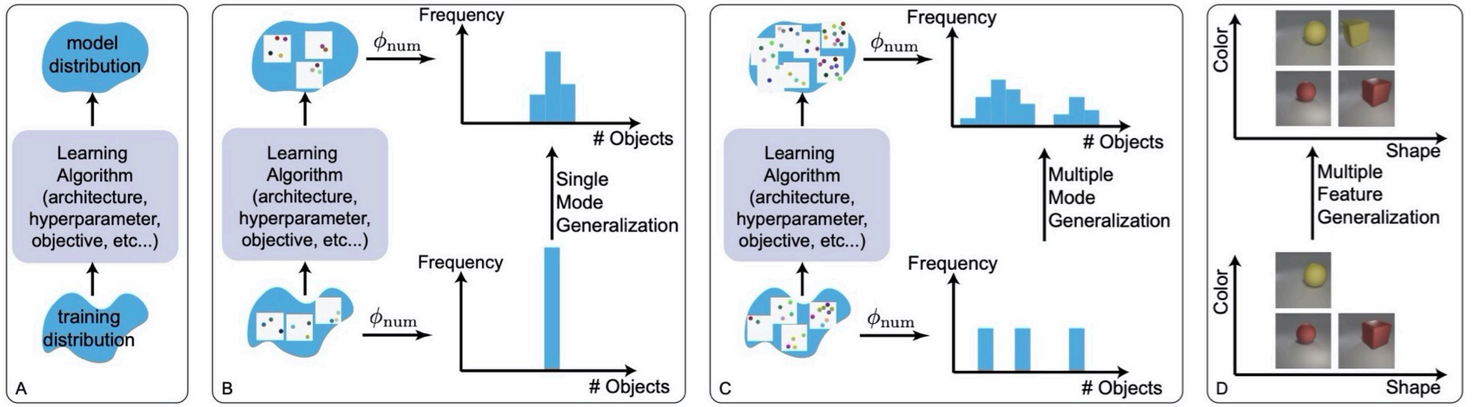
Top: HYPE scores of different models over CelebA and FFHQ datasets. A score of 50% represents indistinguishable results from real, while a score above 50% represents hyper-realism

Bottom: Example images sampled with the truncation trick from StyleGAN trained on FFHQ dataset. Images on the right have the highest HYPE scores (i.e. exhibit the highest perceptual fidelity)

Zhou et al., 2019

Measures that Probe Generalization

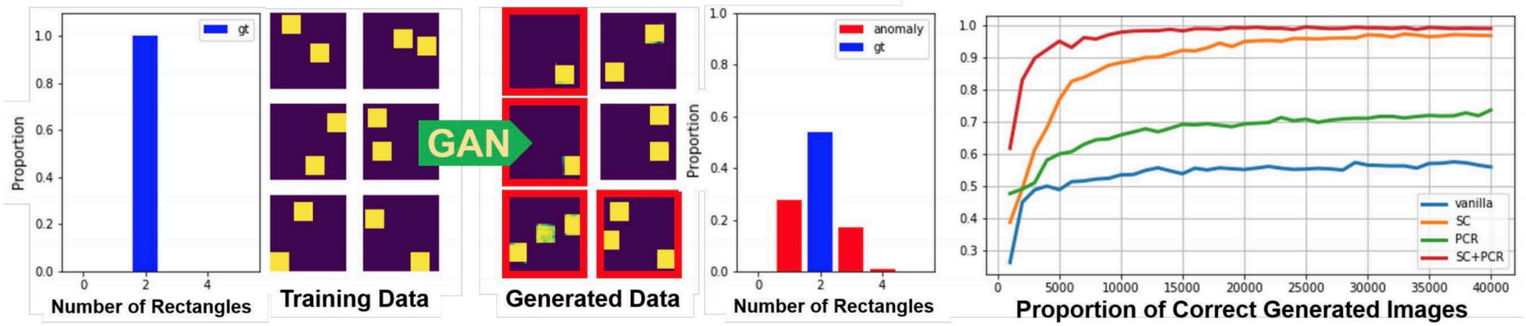
- Attempt to measure the generalization of generative models at the semantic level (e.g., number of objects in the image, relationships among object parts, etc)



A) A generative model can be probed with carefully designed training data. Examining the learned distribution when training data B) takes a single value for a feature (e.g. all training images have 3 objects), C) has multiple modes for a feature (e.g. all training images have 2, 4 or 10 objects), or D) has multiple modes over multiple features

Zhao et al., 2018

- Can be used to study mode collapse and mode drop (more on this later)

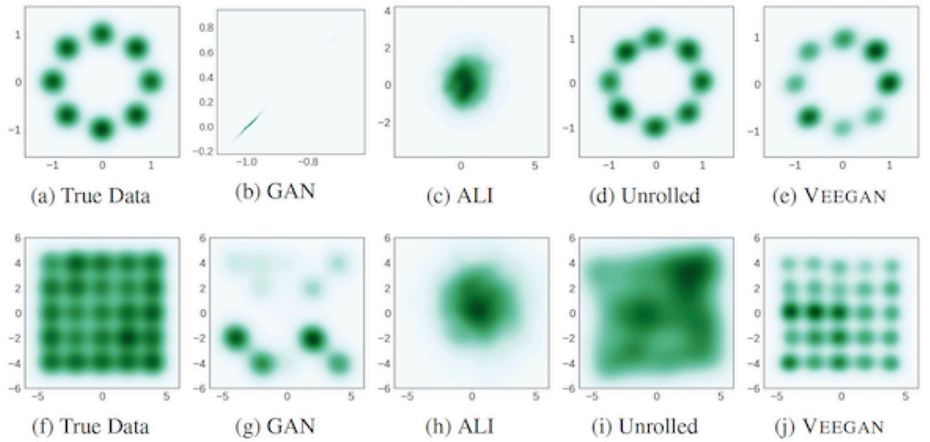


Xuan et al. [48] show that training a GAN over images with exactly two rectangles results in a model that generates one, two, or three rectangles (anomalous ones shown in red)

Xuan et al., 2019

Evaluating Mode Drop and Mode Collapse

- **Mode collapse vs. Mode drop**
- Detecting mode collapse on large scale image datasets is challenging. It is easier on synthetic datasets where the true distribution and its modes are known (e.g. Gaussian mixtures)
- **Santurkar et al. 2018:**
  - Train a GAN unconditionally (without class labels) on the chosen balanced multi-class dataset D
  - Train a multi-class classifier on the same dataset D
  - Generate a synthetic dataset by sampling N images from the GAN. Then use the classifier trained above to obtain labels for this synthetic dataset



Density plots of the true data and generator distributions from different GANs trained on mixtures of Gaussians arranged in a ring (top) or a grid (bottom)

Srivastava et al., 2017

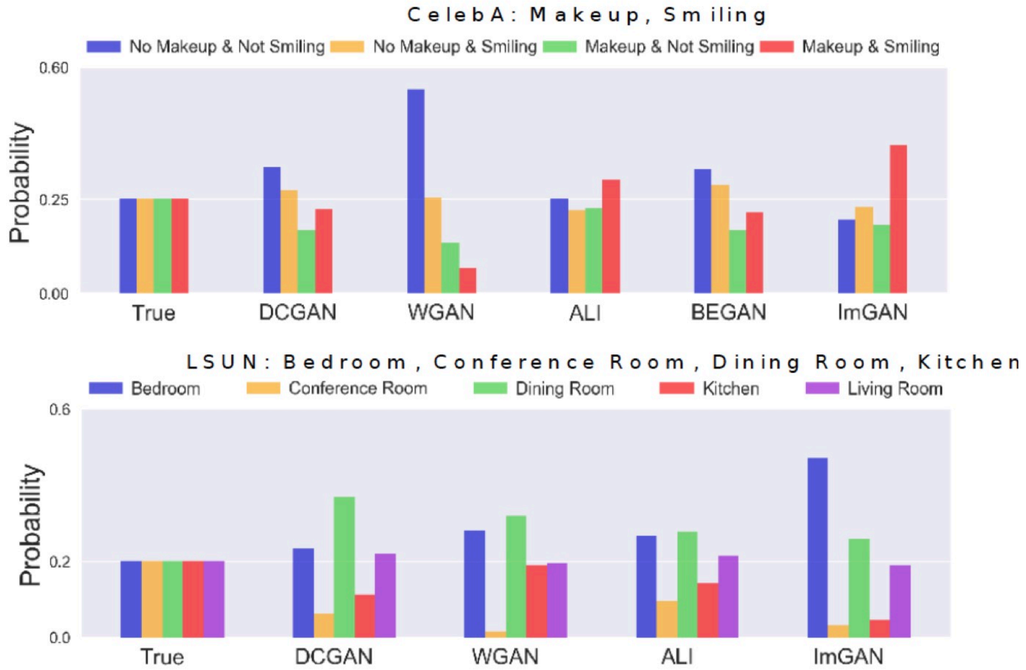
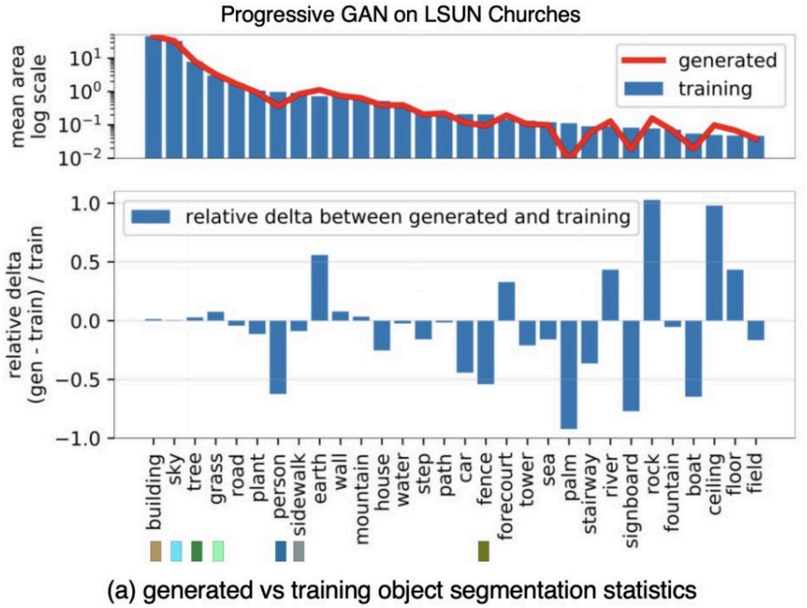


Illustration of mode collapse in GANs trained on subsets of CelebA and LSUN

Santurkar et al., 2018

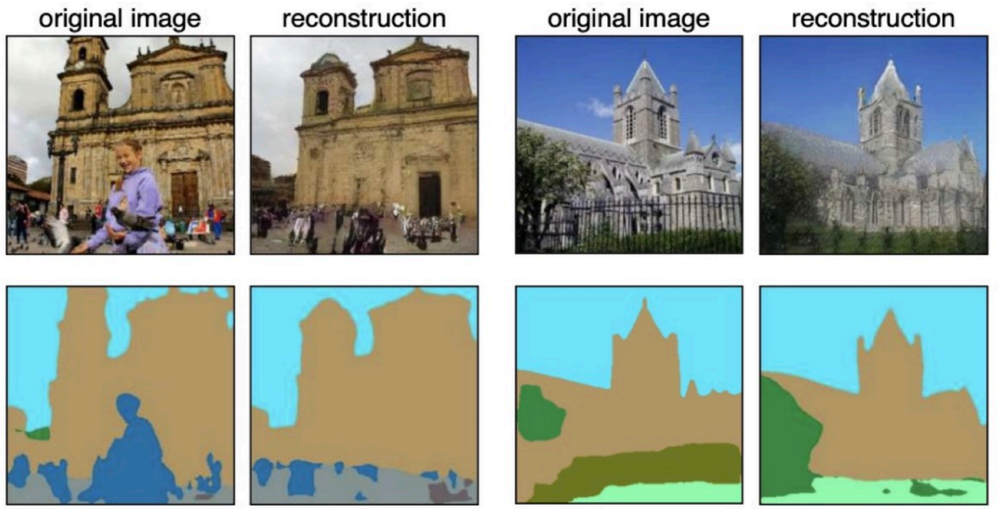
GAN Dissection

- Seeing what a GAN can not generate (similar to FCN score)
- A technique to dissect and visualize the inner workings of GANs
- Identifies GAN units (i.e. generator neurons) that are responsible for semantic concepts and objects (such as tree, sky, and clouds) in the generated images
- Also allows finding artifacts in generated images



(a) generated vs training object segmentation statistics

Distribution of object segmentations in the training set of LSUN churches vs. the corresponding distribution over the generated images. Objects such as people, cars, and fences are dropped by the generator.



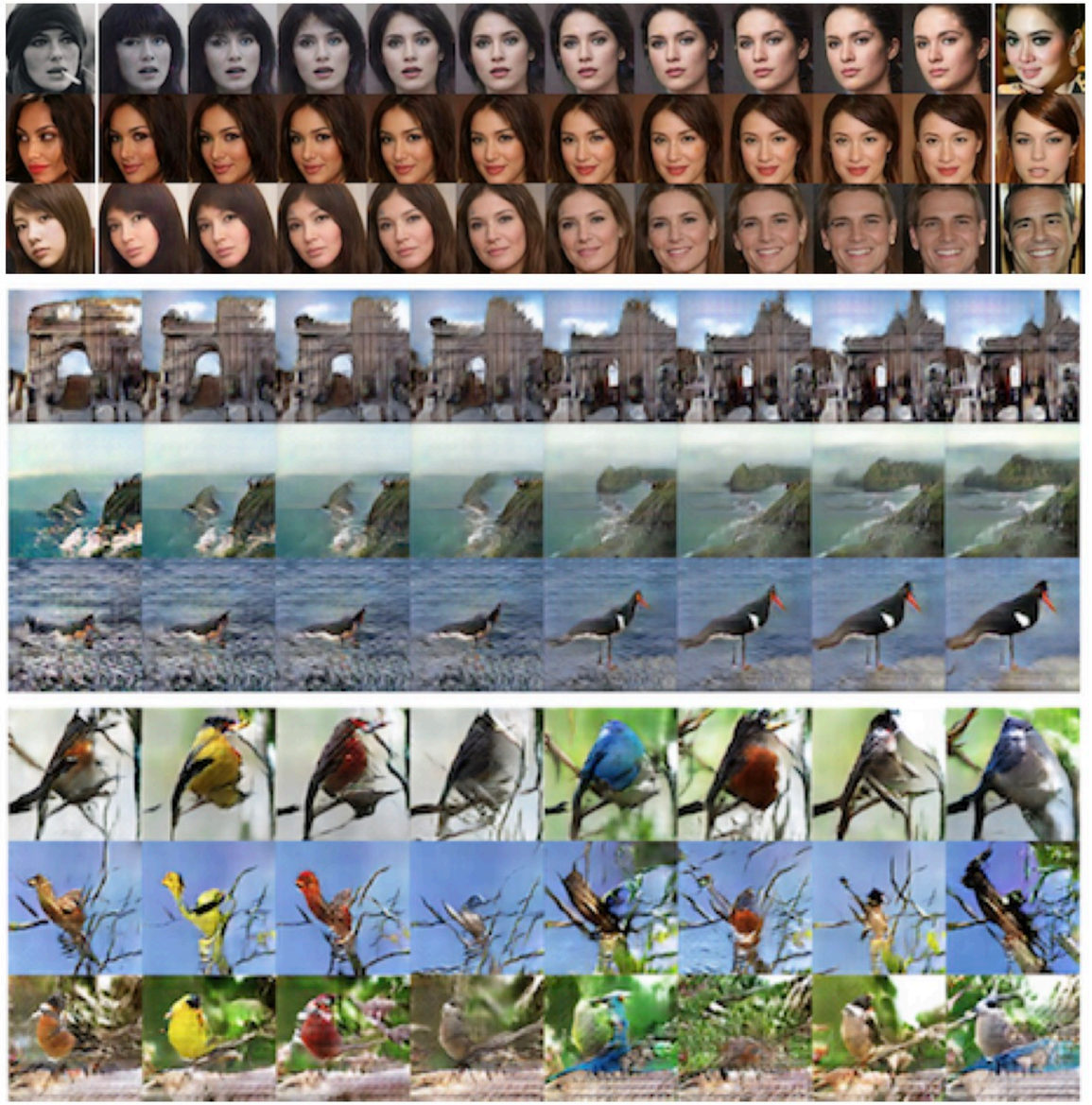
(b) real images vs. reconstructions

Pairs of real images and their reconstructions in which individual instances of a person and a fence cannot be generated.

Bau et al., 2019
https://gandissect.csail.mit.edu/

Investigating and Visualizing the Internals of Networks

- Disentangled representations. Alignment of “semantic” visual concepts to axes in the latent space
- Space continuity
- Visualizing the discriminator features
- PPL, GAN Steerability
- Useful for image editting



GAN Steerability

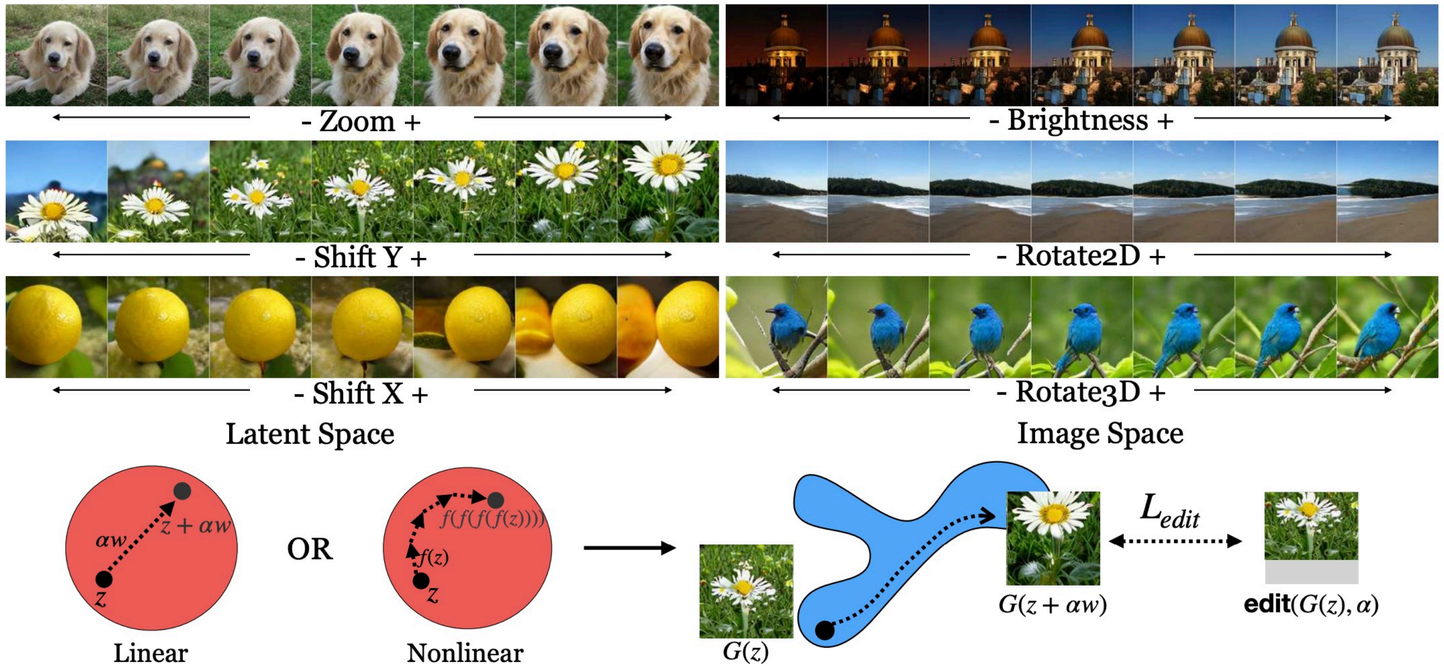
- Quantifies the degree to which basic visual transformations are achieved by navigating the latent space of a GAN
- The goal is to find a path in z space to transform the generated image G(z) to its edited version edit(G(z, α)), e.g. an α× zoom
- To measure steerability, the distributions of a given attribute in real images and generated images (after walking in the latent space) are compared

The task it to learn the walk w by minimizing:

$$w^* = \texttt{argmin}_w \, \mathbb{E}_{z,\alpha}[\mathcal{L}(G(z+\alpha w), \texttt{edit}(G(z), \alpha))],$$

where α is the step size, and L is the distance between the generated image G(z + αw) after taking an α-step in the latent direction, and the target image edit(G(z), α)

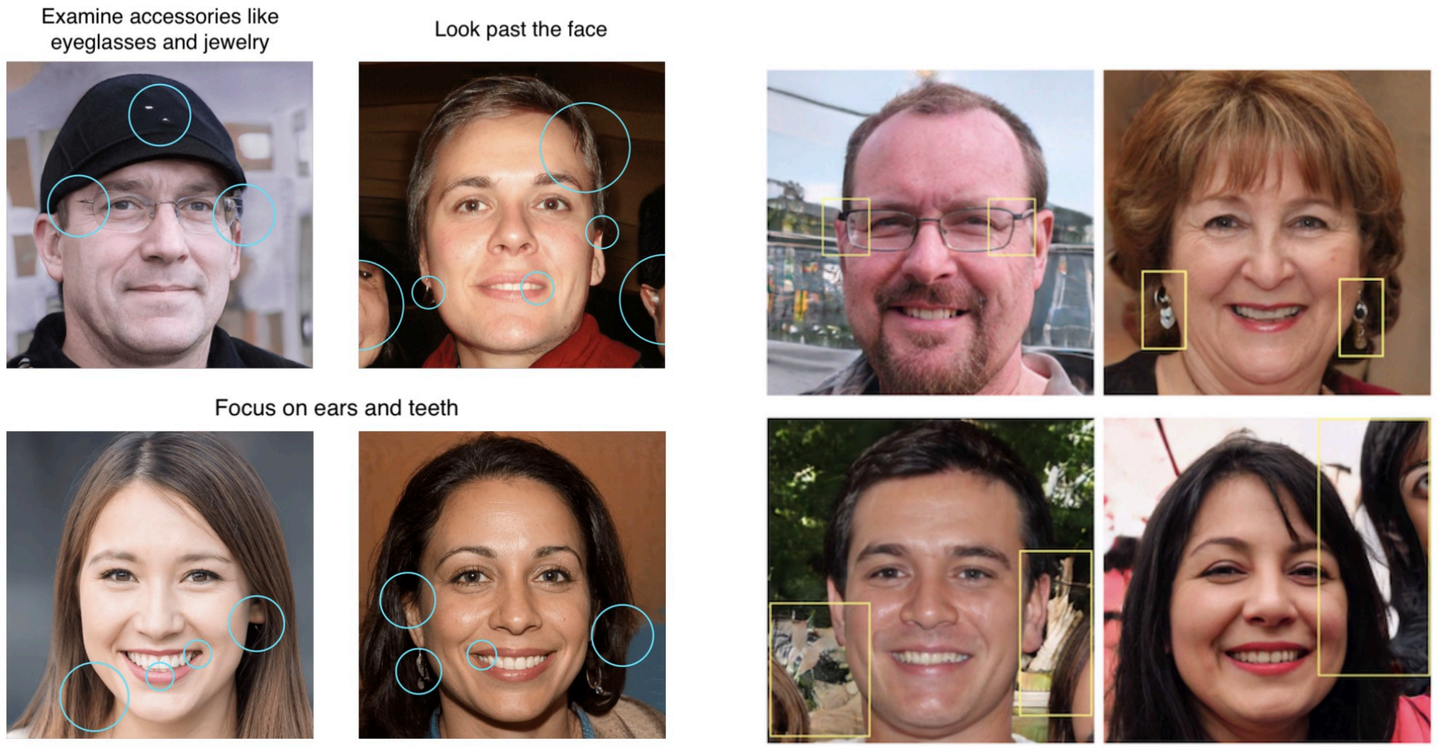
A walk in the latent space of a GAN corresponds to visual transformations such as zoom and camera shift



Jahanian et al., 2019
https://ali-design.github.io/gan_steerability/

Connection to Deep fakes

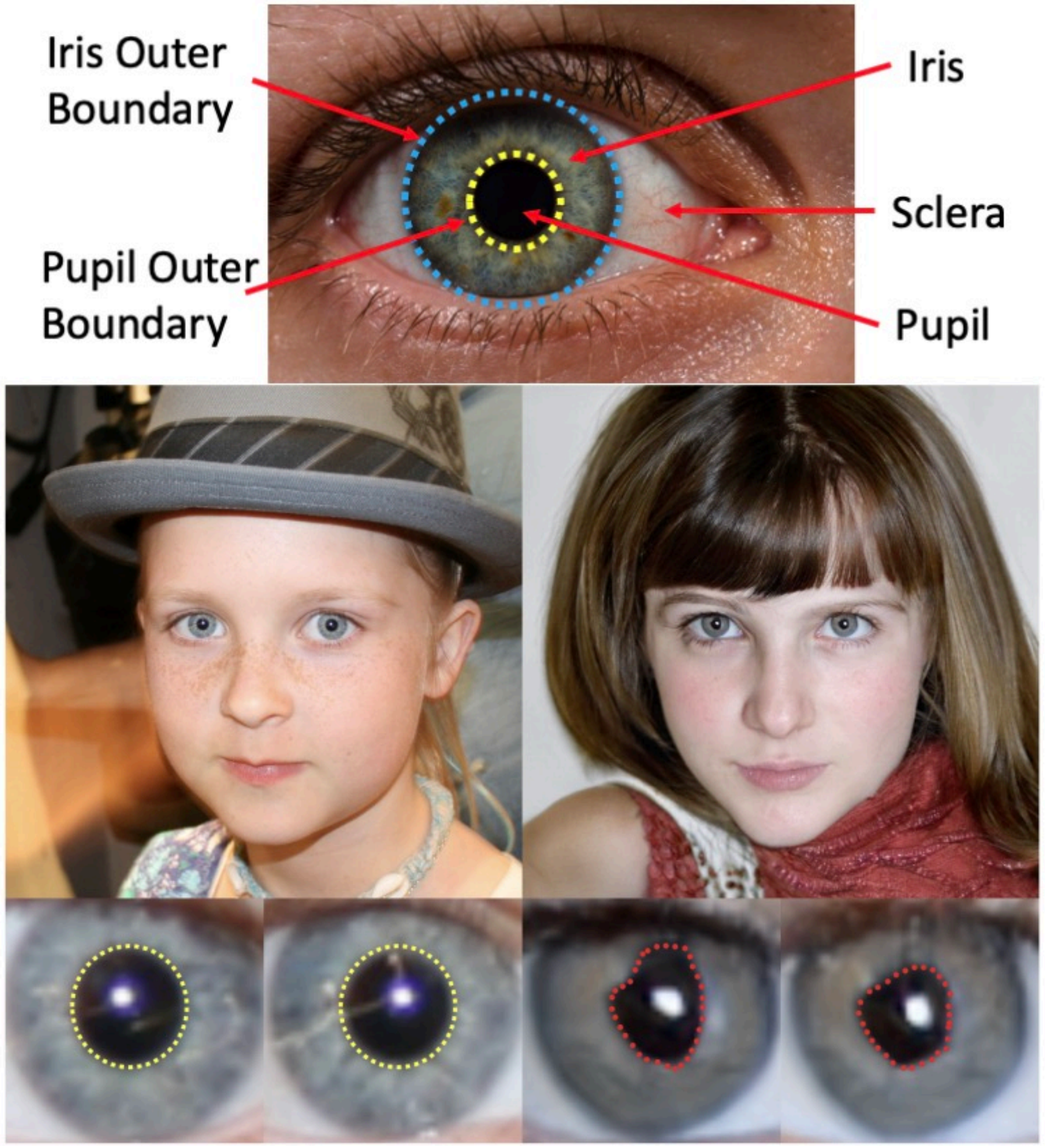
- A alarming application of generative models is fabricating fake content
- Natural connection between deep fakes and GAN evaluation
- Difficulty of deepfake detection for humans is category dependent. Some categories such as **faces, cats, and dogs being easier than bedrooms or cluttered scenes**



Sample generated/fake faces and cues to tell them apart from real ones. See also https://chail.github.io/patch-forensics/. Image courtesy of Twitter.

Eyes Tell All:
Irregular Pupil Shapes Reveal Gan-Generated Faces

Guo et al., 2021



Top: Anatomy structures of a human eye.

Bottom: Examples of pupils of real human (left) and GAN-generated (right). Note that the pupils for the real eyes have a strong circular or elliptical shapes (yellow) while those for the GAN generated pupils are with irregular shapes (red). And also the shapes of both pupils are very different from each other in the GAN-generated face image.

Guess Game!

- Can you tell which face in a pair is real?

https://www.whichfaceisreal.com/.

Other Categories

https://thiscatdoesnotexist.com

https://thisrentaldoesnotexist.com/

See also https://thisxdoesnotexist.com/. Although images look realistic in the first glance, a closer examination reveals the artifacts

Summary

- GAN evaluation is still an unsolved problem
- Some measures such as IS, FID, PPL, and PR are more accepted
- Other measures:
  - Based on statistics of natural scenes
  - Geometry score
  - Boundary distortion
  - …
- Other aspects and dimensions for comparing measures:
  - Sample efficiency
  - Computational efficiency

Future

- Extension to other domains such as text, video, audio, and tabular data
- Benchmarking measures and models
- Relation to adversarial examples
- Relation to generalization
- Relation to deep fakes
- Bias and fairness
- …

References

- https://github.com/xuqiantong/GAN-Metrics
- https://www.coursera.org/specializations/generative-adversarial-networks-gans
- https://machinelearningmastery.com/how-to-evaluate-generative-adversarial-networks/
- https://jonathan-hui.medium.com/gan-how-to-measure-gan-performance-64b988c47732
- https://github.com/google/compare_gan
- https://github.com/geek-ai/Texygen
- https://paperswithcode.com/sota/image-generation-on-cifar-10
- https://towardsdatascience.com/graduating-in-gans-going-from-understanding-generative-adversarial-networks-to-running-your-own-39804c283399
- https://analyticsindiamag.com/top-6-metrics-to-monitor-the-performance-of-gans/
- https://github.com/open-mmlab/mmgeneration
- https://github.com/Diyago/GAN-for-tabular-data
- https://github.com/mbinkowski/MMD-GAN

KEEP
CALM
AND
STAY
SAFE